DOCUMENT RESUME

ED 325 040 HE 023 958

AUTHOR	Finlayson, Jean, Ed.
TITLE	Management Information Systems.
INSTITUTION	Further Education Staff Coll., Blagdon (England).
REPORT NO	ISSN-0305-8441
PUB DATE	89
NOTE	96p.
PUB TYPE	Collected Works - General (020) -- Guides - Non-Classroom Use (055)
JOURNAL CIT	Coombe Lodge Report; v21 n4 p225-318
EDRS PRICE	MF01/PC04 Plus Postage.
DESCRIPTORS	*College Administration; *Computer System Design; *Evaluation Criteria; Foreign Countries; Higher Education; Information Management; *Management Information Systems; Selection; Staff Development; *Systems Development; Training; *User Needs (Information)
IDENTIFIERS	*England

ABSTRACT

This collection of papers addresses key questions facing college managers and others choosing, introducing, and living with big, complex computer-based systems. "What Use the User Requirement?" (Tony Coles) stresses the importance of an information strategy driven by corporate objectives, not technology. "Process of Selecting a Computerised MIS in Cheshire" (Dave Plunkett) emphasizes a criterion-based process within a model of rational decision making. "Choosing a Management Information System at Hounslow Borough College" (Ian Harrison) describes more of the selection process in terms of its human impact. "MIS: An Introduction Strategy" (David Atkinson) distinguishes between "implementation" which is defined as largely a technical process, and "introduction," which concerns itself more with the human factors that can thwart the development and use of a system. In "Living with a Management Information System," Jean Skuse and Robert Case report their experiences with particular suppliers. "Staff Development and Training for Management Information Systems" (Paul Crisp) notes that all staff development and training needs can rarely be met and suggests the kinds of training that are likely to be considered essential. "FESC Guide to Computer Jargon" (Paul Crisp and others) interprets obscure jargon from the world of computers that is now penetrating education management. References accompany each paper. (JDD)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED325040

Management Information Systems

SCOPE OF INTEREST NOTICE

The ERIC Facility has assigned this document for processing to:

In our judgment, this document is also of interest to the Clearinghouses noted to the right. Indexing should reflect their special points of view.

HE

Coombe Lodge Report

HE023 958

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Further Education Staff College

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☐ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

The Further Education Staff College

COOMBE LODGE REPORT
Volume 21 Number 4

MANAGEMENT INFORMATION SYSTEMS

By David Atkinson, Robert Case, Tony Coles, Paul Crisp, Mick Fletcher, Ian Harrison, Dave Plunkett, Dave Senior and Jean Skuse

Coombe Lodge Reports

Publications Editor: Jean Finlayson

Publications Administrator: Philippa Toogood

Clerical Assistant: Sarah Tanner

The views expressed in this report are those of the contributor(s). They should not be taken to represent the policy of The Further Education Staff College.

Published by The Further Education Staff College
Coombe Lodge, Blagdon, Bristol BS18 6RG
Telephone (0761) 62503

Typesetting by Avonset, Midsomer Norton, Bath
Printed by Colourways, Clevedon, Bristol

ISSN: 0305-8441

© The Further Education Staff College 1989

All rights are reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, electrical, chemical, optical, photocopying, recording or otherwise, without the prior permission of the copyright owner.

MANAGEMENT INFORMATION SYSTEMS

Contents

FOREWORD
Jean Finlayson
Publications Editor, Further Education Staff College 225

WHAT USE THE USER REQUIREMENT?
Tony Coles,
Principal, FHE2
Department of Education and Science:
User Requirement Project Team Leader 229

PROCESS OF SELECTING A COMPUTERISED MIS IN CHESHIRE
Dave Plunkett
MIS Development Officer
Cheshire LEA 237

CHOOSING A MANAGEMENT INFORMATION SYSTEM AT HOUNSLOW BOROUGH COLLEGE
Ian Harrison
Principal Administrative Officer
Hounslow Borough College 245

MIS: AN INTRODUCTION STRATEGY
David Atkinson
Staff Tutor
Further Education Staff College 253

LIVING WITH A MANAGEMENT INFORMATION SYSTEM
Jean Skuse
CMS Co-ordinator and
Robert Case
Education Information Systems Officer
Berkshire County Council 263

STAFF DEVELOPMENT AND TRAINING FOR MANAGEMENT INFORMATION SYSTEMS
Paul Crisp
Staff Tutor, Further Education Staff College 269

FESC GUIDE TO COMPUTER JARGON
Paul Crisp
Staff Tutor, Further Education Staff College,
Mick Fletcher
Staff Tutor, Further Education Staff College and
Dave Senior
Formerly Computer Manager, Further Education Staff College 285

FOREWORD

This Coombe Lodge Report is the first to concentrate wholly on Management Information Systems (MIS). It reflects the current high level of interest in MIS within the post-statutory sector. The papers gathered here address some of the key questions facing college managers and others choosing, introducing and living with big, complex computer-based systems.

FESC's interest in MIS goes back to the mid-'70s when the college proposed and then built the first MIS tailored specifically for the UK further education system[1]. For some time FEMIS[2] and the CMS[3] package were the two major college-based information systems, but usage (if not ownership) of the systems was patchy. The most widespread were, in fact, the ones operated by LEAs — designed primarily, in some cases solely, to produce the FE statistical return for the DES. When, in 1985, the Audit Commission turned its attention to FE it declared the service wanting. One of the things it wanted was better information. The Joint Efficiency Study's (JES) report, **Managing colleges efficiently**, was more specific. Every college, it said, 'should be operating comprehensive, integrated, computerised management information systems . . . within five years'. To assist progress in this direction, the DES established a study group to propose some national guidelines which were published as the CMIS User Requirement, referred to in Tony Coles' paper.

The JES report and the CMIS User Requirement, coupled with the emerging consequences of the Education Reform Act, have generated an enormous amount of activity around MIS. Few now need convincing of the need for an MIS in their colleges and LEAs. This Coombe Lodge Report is therefore sparing of evangelism. It focuses instead on the issues and

1 Then the FEMIS Research Project, now one of the products of EMIS Ltd.
2 Further Education Management Information System.
3 College Management Systems produced by The Fretwell-Downing Computer Group.

concerns which follow the decision to invest in MIS. Tony Coles begins by giving his view from the centre of the future for college MISs. He stresses the importance of an information strategy driven by corporate objectives — not technology. Nevertheless, he notes, the technology continues to evolve and promises to offer new and better ways of managing and using information.

David Plunkett's paper describes the method used by Cheshire LEA to select an MIS as a systems approach. The criterion-based process he sets out is a model of rational decision making. Ian Harrison starts from a similar position but describes more of the process in terms of its human impact.

David Atkinson asks us to distinguish between 'implementation' which he suggests is largely a technical process, and 'introduction', which concerns itself more with the human factors which can thwart the development and use of MISs. He analyses the sources of resistance to the introduction of an MIS and notes that these can exist in management as well as in staff. A plan is needed, he proposes, which takes account of these factors.

Jean Skuse and Robert Case write from the perspective of the implementer and co-ordinator and offer some insights into the burdens of living with the system once introduced. They are necessarily reporting their experiences with particular suppliers (as, indeed, are Plunkett and Harrison) but we can be confident that a similar story could be told with any other. Paul Crisp looks at the issue of staff development and training from the point of view of different MIS–related roles and functions. Although taking the ideal model as a starting point, he notes that we can rarely meet all the staff development and training needs we identify and suggests the kinds of training which are likely to be considered essential.

The Report concludes, not with another paper, but with a guide — The FESC Guide to Computer Jargon. The compilers, Crisp, Fletcher and Senior, have collected many of the obscure jargon terms from the world of computers now penetrating education management, and try to interpret them for us. We are warned not to take all their explanations completely seriously!

Together, these papers offer useful insights and timely, practical advice to those in colleges and LEAs now moving into the difficult and obscure world of computer information systems. It is, perhaps, reassuring for the

non-technical among us that several of the authors stress the human management issues over the technological. And if they do use some technical terms, well, we can always look them up in the Guide.

Jean Finlayson
Publications Editor
Further Education Staff College

WHAT USE THE USER REQUIREMENT?

Tony Coles
Principal, FHE2
Department of Education and Science:
User Requirement Project Team Leader

This paper gives a perspective from within the Department of Education and Science (DES) on management information system developments in general and the implementation of the *User requirement report* in particular.

BACKGROUND

In 1985 the Audit Commission complained that the absence of useful management information had sometimes made it impossible to make any judgements about the use of college resources. Two years later, the Joint Efficiency Study (JES), in **Managing colleges efficiently**, estimated that at least half the colleges had a computerised management information system (CMIS) of one kind or another, but concluded that not even the most sophisticated system then available was capable of producing the information requirements implicit in its recommendations for improving college management. To define the information needs of local education authority (LEA) and college managers at all levels, the DES set up a project team to produce a user requirement. The project team's report, **User requirement for computerised management information systems for further education colleges**, was published in July 1988. Copies have been distributed to every LEA and further education (FE) college.

THE USER REQUIREMENT REPORT

The purpose and scope of CMISs

Management is about change. Anxiety about change is not something peculiar to the 1980s, it is endemic to the human condition. Plato decided that it was synonymous with decay; Machiavelli thought there was nothing more difficult, more doubtful of success, or more dangerous than initiating change. Darwin, however, saw that while change was potentially life-threatening, those who responded well to its challenges not only survived but went from strength to strength. The lesson from Natural Selection is that like it or not, life behoves us all to respond to change, to embrace evolution. Translated to the world of FE this means that to survive, colleges have to respond to changing client needs, which are in turn a function of socio-economic, technological, demographic, curricular and pedagogic change. To thrive and prosper colleges will have to manage their resources more efficiently and effectively than in the past. This will require managers to adopt a more pro-active approach to resource management. They will have to plan ahead, anticipate developments and react effectively to change as it occurs.

This is where a CMIS comes in: its purpose is to facilitate the positive management of change. And purpose is all: for unless the CMIS provides the information necessary to stimulate early adjustments to day-to-day operations and prompt critical appraisals of long-term plans, its outputs lie unused upon managers' desks.

Strategy

Strategy is where it begins and ends; get the strategy right and the rest will follow. Which means that the over-all strategy must be business, not technology driven. It must take a holistic view and use technology to enhance corporate efficiency and effectiveness. Above all the strategy must ensure that the CMIS is fully integrated into a responsive organisational structure, which is itself supported by sound managerial practices. This can best be accomplished by an authority-wide Information Technology Strategy Group which should take account of where the authority is and where it wishes to go in organisational and data management terms. In other words, to be effective the information system ought to form an integral part of corporate thinking, and managers, from the top down, ought to have a clear idea of how the system can serve their needs. In practice, this can best

be achieved by a hierarchy of management plans which distribute responsibilities to specific post-holders and grades, and which clearly identify the aims and objectives of each. By this means CMIS outputs can be directly related to specific objectives.

Appendix C of the User requirement report sets out a series of schedules which, for each particular managerial area, relates responsibilities and objectives to CMIS outputs.

The User requirement report acknowledged that it would be impracticable to develop and implement all in one go a system capable of generating all the information it identified. But starting with the basic processes and phasing in additional, more complex applications over time will help ensure that the strategy is affordable and manageable, and that its benefits can easily be assimilated. To quote from an Education Support Grant (ESG) bid:

It is the authority's view that it was right to explore to the fullest extent the . . . use of micro-computers for the first phase . . . For a comparatively low unit cost it has been possible to spread computer literacy throughout the colleges as a step towards further development . . . The colleges (now) require . . . multi-user, large capacity mini-computer systems.

Finally, the report concluded that it would be wrong to establish a three-to-five year strategy and leave it at that. Business needs will change, new organisational structures may evolve, and technology marches on. The over-all strategy, and the attendant strategies for data-capture, networking, hardware, software and training, should therefore be reviewed systematically and updated.

Implementation

The User requirement report sets out a broad national framework. Its implementation in a particular college or LEA will require further assessment of local needs. LEAs will consequently have to give proper consideration to their hardware, network and software requirements, data-capture strategy and training needs. To implement the decisions of the Strategy Group, authorities should consider setting up a central unit which, besides providing systems support, could also carry out O & M reviews. This would help engender a positive interaction between the MIS and organisational evolution.

Hardware and software

The report sets out a suggested order for the phased introduction of the various software modules and associated files. There are seven phases in all. Completion of the first four should enable a CMIS to generate all the JES indicators and CIPFA unit costs, and meet central government's information needs. As for hardware strategy, LEAs and colleges can choose one of two routes. They can opt for a single supplier which offers a range of computers with consistently implemented operating systems. Alternatively, they can adopt a strategy which focuses on emerging international standards such as UNIX operating systems for multi-user computers. Both strategies provide their own particular benefits; the first offers a single (and cheaper?) requirement for training, maintenance and development; the second provides flexibility and greater freedom of choice.

The DES is not committed to any one particular software supplier. Indeed, since the issue of the **User requirement report** the department has had discussions with several major companies. The department is also keen that the many sub-systems that are developed by colleges and individual departments should not go unnoticed. To this end, we are encouraging software suppliers to seek out such sub-systems, develop them to industry standards and market them vigorously. By this means we hope to advance the User requirement's integrated approach to sub-systems, obviate the re-invention of the wheel, ensure compatibility with other systems and provide better customer support.

Networks and data-capture

To take proper advantage of the JES and the Education Reform Act's planning and delegation proposals, college staff will require interactive, on-line access to the college's main MIS, and LEA and college staff will want similar access to each others' systems. So whatever the hardware strategy, the hardware development path must be able to support local and wider area networking. Given DES Circular 9/88's call for LEAs 'to ensure proper co-ordination between provision in FHE colleges and 16-19 provision in schools and sixth-form colleges', the interaction between college databases and those of the local authority's education department and treasurer's office need not be the last word; the net could be spread wider. As one LEA's ESG bid had it:

The colleges' MIS should not be seen as a separate entity, and consideration must be given to the relating of data from the Schools Sector, Adult and Continuing Education Service and the Careers Service.

That said, a 'no hiding place' network which, in the words of one LEA, would give 'unfettered access' to college databases, should not be used by LEAs as an excuse for excessive interference in a college's day-to-day affairs.

Properly used, networks that embrace all the above databases can inform post-16 strategic planning, and support the User requirement's recommended data capture strategy: which is to capture data only once and as close to source as possible, optimise the sharing of data between systems, and maximise value-for-money by making it easy to access and manipulate.

Training

In untrained hands a computer terminal can become an instrument for destruction. Careless data inputting can corrupt files, produce nonsensical outputs and destroy confidence in the system as a whole; while the unskilled manipulation of data can at best frustrate, and at worst produce specious results. It is therefore imperative that the introduction of a CMIS should be accompanied by a series of training programmes which have been carefully tailored to meet the needs of the various users; the needs of operators being clearly different from those of the principalship. Operators will be preoccupied with data entry and retrieval, while the principalship will be more concerned with mastering decision support systems. The phasing of training programmes should parallel the phased implementation of the CMIS.

DISASTER PLANNING

Today's computer installation with its attendant network of terminals is an extremely sophisticated, but highly vulnerable structure. Terminals can malfunction, the network be disrupted, and the central processing unit be destroyed by fire, flood or vandalism. As LEAs and colleges become more computer-dependent the consequences of disaster grow. But, if ESG bids are anything to go by, very few LEAs have given much thought to disaster planning.

If CMISs have been properly designed and backed up by successful training programmes, the loss of computing power for more than a day or two could cause severe withdrawal symptoms. It is therefore vital that LEAs and colleges should produce effective disaster recovery plans. Whether an LEA or a college goes in for insurance policies, such as 'cold start', i.e. the guaranteed provision of temporary accommodation in which to install their replacement equipment, or 'warm start', i.e. the guaranteed immediate delivery of replacement equipment for installation in their premises until normal procurement has been completed, or simply relies on distributed processing and/or common systems, is not just a matter of finance; it is also a matter of an analysis of perceived risks and the consequences of long-term loss of computer power.

COSTS AND BENEFITS

Costs

The cost of a CMIS based on the User requirement report, with all its implications for the size and nature of the database and its assumptions about local and wider area networking, will range from £50,000 to £150,000 per college, depending on the size of the college, its structure, and whether it is single or multi-site. To support such expenditure, ESG funding under the Department's three-year programmes is providing the equivalent of £70,000 per college. In addition, the Training Agency's Mutual Development Fund is supporting expenditure for developing and implementing an MIS strategy. This central government funding is, of course, in addition to the very large sums many LEAs have already spent from their own resources.

Benefits

Replacing a plethora of manual and stand-alone computerised systems with a single integrated CMIS should result in savings. One LEA has estimated that the total annual cost of manual systems to produce the outputs identified in the User requirement report would be in the order of £120,000 to £180,000 for its five colleges. The same LEA has also estimated that a one per cent improvement in the utilisation of teaching staff and a five per cent increase in colleges' income resulting from introducing CMISs, would generate benefits in excess of £300,000 per annum. To these factors one

could add others, such as the benefits resulting from improvements in the use of accommodation, for inclusion in any cost-benefit analysis of CMISs.

CONCLUSIONS

Much has already been accomplished in the field of CMISs, but progress is by no means uniform. Some LEAs and their colleges are already in sight of the sort of integrated system proposed by the User requirement report; others have barely left the starting blocks. Some have firmly grasped the concept of a systemic approach to a CMIS and the system's relationship to corporate planning and information management; others still cannot distinguish between an administrative system and a management information system.

There is no one system or single development path that will provide for every LEA's or college's particular needs. However, the User requirement report provides an outline of what the end product should look like and how it might be achieved. Implementing a CMIS based on the report will not be cheap, nor will it be easy; and it will not be accomplished overnight. A CMIS cannot perform miracles, it can't turn straw into gold, but used to good effect it can provide tangible benefits for staff and students alike. And above all, it can provide the wherewithal to respond to the challenges of change. Remember dinosaurs!

REFERENCES

Audit Commission (1985) **Obtaining better value from further education.** HMSO

Education & Science, Department of (1988) **Education Reform Act 1988: local management of further and higher education colleges: planning and delegation schemes and articles of government.** DES Circular 9/88

Education & Science, Department of & Welsh Office (1988) **User requirement report for computerised management information systems for further education colleges: a report of a study funded by the Department of Education and Science.** DES/WO

Joint Efficiency Study (1987) **Managing colleges efficiently: Report of a study of efficiency in non-advanced further education for the DES/WO and the Local Authority Associations.** DES/WO

PROCESS OF SELECTING A COMPUTERISED MIS IN CHESHIRE

Dave Plunkett
MIS Development Officer, Cheshire LEA

This paper covers the seven colleges operating in Cheshire, six of which are FE colleges plus one specialist college of agriculture. Cheshire's colleges entered the 1980s using the IBM mainframe system at County Hall to produce invoices and the FESR. Student enrolment data were transmitted from college-based terminals, the results aggregated and a tape dispatched to the DES by the mainframe. Colleges could also use the terminals to access the mainframe FISC system for accounts payable.

EARLY SYSTEMS

With the increased availability of micro-computers during the early 1980s, colleges began to develop their own systems using Lotus, dBase and similar packages which were responses to college management requests for more standardised means of producing management information. In early 1984 a small group was established to evaluate the FEMIS package which North Cheshire College had already implemented. Following the study recommendation, FEMIS was purchased for all colleges and was implemented in September 1986. In parallel developments colleges had also begun to use the FEU Employers Needs Database package for recording employers' requirements and visits to employers. These systems were based on IBM PCs. Cheshire had begun to develop the idea that all colleges would use the same computers and systems so as to allow common aggregation of data by the LEA. This approach has continued and been strengthened in the last few years with the developments described in this paper.

During 1986 the LEA and the colleges became increasingly aware of several simultaneous developments: the new FESR requirements for 1988; the requirements of the Student Information Systems Pilot (TC, 1987); the emerging intention of the JES (DES/WO, 1987) following the Audit Commission (1985) study of FE; and also the realisation that the mainframe system described above could not be adapted to meet these growing requirements. In late 1986 a feasibility study was mounted to look at the alternatives.

FEASIBILITY STUDY

The feasibility study produced at the end of 1986 examined three options. The first would be to write our own mainframe system using our Management Services Department at County Hall. The second was the new FEMIS student record module. The third was the Fretwell-Downing CMS. The study considered the first, but pointed towards the comparison of FEMIS and Fretwell-Downing in detail as being the only realistic solution whereby systems would be centred in colleges and allow colleges to use them to produce useful operational outputs as well as deriving substantial management information from their control of the systems.

Accordingly, in early 1987, a small team was set up to take a closer look at the two packages identified. This team comprised representatives of the LEA and colleges including a vice principal, two PAOs, computer operators and staff from Management Services Department, with Sue Jones writing the report.

The process of evaluating hardware and software packages is quite standard and several textbooks defined the sort of processes that a prospective user should use to evaluate such packages; Beaman (1984) has produced a useful guide for a small business in selecting packages. The methodology of our study was to contact both suppliers and, following visits to the companies and as far as possible to users, to compare the findings against the evaluation criteria.

EVALUATION CRITERIA

Our earlier FESR feasibility study had stated that either system would meet the requirements of colleges for providing information for themselves and for the LEA, DES, TA, etc. They each promised to provide a range of

functions which would build up into a complete management information system. Colleges would implement such a system in phases and it was recognised that they would therefore be dependent on the supplier for support, day-to-day maintenance and future developments.

The evaluation criteria decided upon by the team were as follows.

1 **Usability for the operator** Is it easy to find your way through the system, are the menus helpful, do the screens have a consistent style? Are help facilities provided? Can you go directly to the programme you want once you have become familiar with the system? Are standard look-up tables provided with the system or do they have to be entered by the college? Will these tables be up-dated automatically?

2 **Usability by the manager** How many reports come as standard with the system? To what extent is it left to the manager to identify useful reports and gain experience and become familiar with the system? Does the system then lend itself to giving the user some vision of how it may best be used?

3 **File design** Is the system flexible, robust and capable of enhancement and future growth? Will it make the system easy to maintain?

4 **Supplier competence** Has the supplier a good grasp of the working of the college? Is the supplier capable of supporting a large scale development over a long period? Does the supplier recognise the need for support and training and to what level are these provided?

5 **Supplier stability** How long has the supplier been in business? How many products does the supplier produce? How dependent is the supplier on the success of this one product? What are the future plans of the supplier?

6 **Product support** Should the supplier go out of business could the system be supported and/or developed by anyone else?

7 **Support within Cheshire** How much support and training would be required from within Cheshire from the Management Services Department and from the Education Department?

8 **The ability to add features ourselves** Could we develop programmes to interface with the system? This question should take into account the extent to which we would need to add features to the facilities provided.

9 **Growth potential** Are there limits to growth in the system? How many screens will it support? What is the machine upgrade path?

10 **Communications and other facilities** Will the system be capable of communicating with the mainframe and the viewdata system? What communication standards does it use, for example, SMA, OSI? Could it also run a word processor and a spreadsheet?

11 **Performance** Does the system perform adequately and reliably?

12 **Long and short-term benefits** To what extent would the system provide benefits in the short term? To what extent are the benefits mainly long term?

13 **Compatibility with other Cheshire systems** Are the hardware and software compatible with other Cheshire systems?

Some of these criteria were obviously more important than others. It was not the intention of the review team to produce a weighted ranking assessment. The purpose of this check list was to provide a basis for the decision. The composition of the review team was such that the different needs and opinions of the people involved would allow an over-all impression to come through. We did not therefore score each system and each aspect of the system with marks out of 10 and produce a numerical total that formed the basis for a decision. Both of the packages were evaluated under the headings listed with comments and impressions drawn from all the team members.

METHOD

The evaluation of the Fretwell-Downing system took the form of a two-day visit to the firm's offices in Sheffield. The first day was spent with an overview of the system, modules and the support including documentation

and recommended hardware. The second day was spent in the Training Centre at Sheffield with hands on experience of inputting foundation data and sample student enrolments. Discussion also took place on installing the system and implementation. There was a separate session on how the LEA could expect to use information from its colleges using the system.

For the FSR system, Derek Wickham, from EMIS, and John Fowie, from de Havilland College, came to Chester and gave a demonstration of the system with details of training support and future developments. John Fowie was also able to give a college administration view of how the system operates. This was accompanied by a one-day visit to Alan Bamber at Gloucester College of Art and Technology where the system is run on a Microvax II.

STUDY CONCLUSION

It would be inappropriate here to identify exactly how each of the two systems was evaluated against our criteria. (For colleagues interested in the full report, a copy can be made available.) It is accepted however that different LEAs will have different priorities and that our evaluation of the systems could be different to that of another LEA.

What we think we have achieved is more the *process* of evaluation than the content of *this* evaluation. Using a team allowed a wide range of opinions to be collated. The document which described the evaluation has served as a useful starting point for committees within the LEA and for elected members to see the work that went behind the selection process.

The report identified some of the obvious pressure points that implementing either system would generate. It was stressed that both systems were extremely complex and would require a high degree of commitment from the colleges, the LEA, the County Council Management Services Department and the supplier, if they were to provide the benefits. There would have to be a financial commitment for continued support and maintenance, including up-grade of machinery or modules. The report also identified the managerial commitment which would provide the environment in which the system could be used to best effect. It was an important point throughout the study that these systems were *college management systems*; they were not there solely to produce FESR or other returns to outside bodies. Accommodation and staffing were identified as critical items to be addressed in

colleges, especially as the system developed from a single system replacing student enrolments to a broader system over several years — addressing such areas as students, finance, exams, resources and management information requirements.

The conclusion of the study was in favour of the Fretwell-Downing CMS system which impressed the team with its greater attention to working detail. The team considered that Fretwell-Downing CMS would be easier to implement in Cheshire colleges and to provide more immediate benefits. Because Fretwell-Downing provides a high level of support and training it was thought that there would be less need for a large central support team to be established in the Management Services Department to cope with the technical side of the system. Fretwell-Downing also offered the possibility of a single contract being placed for hardware, software and support.

The FSR system, on the other hand, was not felt to be quite as easy to implement in the short term, although it was suggested that the long-term benefits may be more substantial. But that would require a strong support team either in colleges, at the LEA or our Management Services Department in order to get the best from the system. The seven Cheshire colleges which would use the system are not large colleges with technical staff available on the administration side. It was felt that the FSR and ORACLE would require considerable support from skilled programmers in order to develop the system further. One of the FSR selling points, the adaptability, was in fact considered to be a drawback from the point of view of our colleges. We saw the system as being an administrative and operational tool which would yield management benefits. It has been suggested that polytechnics and larger colleges could support the interfacing requirements that FSR offers and needs, while smaller FE colleges would get more immediate results from Fretwell-Downing CMS at a level more in keeping with their own technical abilities.

It must be noted that the study was conducted during the first half of 1987; a repeat study at a different time or by a different county could produce a different outcome.

OTHER DEVELOPMENTS

The study was completed and published in the summer of 1987 at the time of publication of **Managing colleges efficiently**. A bid under ESG Category

VIII included our 1984 study documentation, our comparative study, and a quotation produced by Fretwell-Downing detailing a three-year implementation plan for the Cheshire colleges. Our ESG bid was discussed with our HMI and gained his support, especially for the fact that the proposal covered all of the colleges in Cheshire and showed how they could progressively implement the modules that would provide the ten performance indicators identified in **Managing colleges efficiently**. Initially three colleges were to commence the installation from spring 1988 onwards, but in January 1988 Fretwell-Downing announced the REFER package which allowed the remaining four colleges to implement a single-user micro computer-based version of CMS in order to produce the new format FESR return for 1988. A CMS implementation group was established, drawing representatives from colleges and Management Services.

In parallel developments during 1987 the LEA has also received a Central Reserve bid in order to respond to the Student Information Systems Pilot Study. This was extended during 1988 to produce a further bid to the Mutual Development Fund for staffing associated with the implementation of management information systems. Colleges can now employ some additional staffing for the three-year duration of the implementation of the Fretwell-Downing modules. My own position as development officer attached to the LEA for three years is also part of the implementation of these systems. My major duty is to assist colleges and establish the interface with the LEA. In year 2 of the project, beginning in spring 1989, the LEA will implement a computer system similar to that of the colleges so that it will be able to collate data for planning purposes, the major aspect of which is the operation of the financing and delegation scheme.

QUALITY AND PERFORMANCE

In Cheshire we have a loose team, comprising FE advisers, FE planning officer and MIS development officer, offering a support service to colleges as they begin to respond to the challenges of **Managing colleges efficiently**. We are now discussing how the development of the 'self-evaluating college' can respond to the issues of quality and performance over the next few years, with performance indicators generated from our CMS systems.

Cheshire, in common with some other authorities, has now firmly established a policy whereby all colleges in the county implement the

same systems. Colleges will obviously implement different modules at different paces and will produce their own version of the system. It is to be expected that college managers and governing bodies will see certain outputs and systems as being more relevant to their college than others.

The LEA's interest is in a standard set of information designed by the college but in line with the expectations agreed by the county, and to some extent suggested by the Joint Efficiency Study. This county approach may well have been a major factor in the position that we have reached so far in terms of successful ESG and Development Fund bids.

FINALLY

We have now had our first year of using the CMS system. It has not been trouble free but then we never expected that it would!

We would recommend a widely drawn team including representatives from the Management Services Department of your county council. Colleges do not always have the technical knowledge which is required. It is evident in our first year of operation that college staff are making great operational use of the system and that is the purpose of the system; we do not wish to develop a technical expertise beyond that necessary for day-to-day operation.

In conclusion I must register my appreciation to the staff from our colleges who are living through our choices, and to the Management Services Department team who have supported us.

REFERENCES

Audit Commission (1985) **Obtaining better value from further education.** HMSO

Beaman, I R (1984) **Small business computers for first time users** NCC (o/p)

Joint Efficiency Study (1987) **Managing colleges efficiently: report of a study of efficiency in non-advanced further education for the DES/WO and the Local Authority Associations.** DES/WO

The Training Commission (1987) **NAFE Student Information Systems Pilot Project: Final Report.** The Training Commission, Sheffield

CHOOSING A MANAGEMENT INFORMATION SYSTEM AT HOUNSLOW BOROUGH COLLEGE

Ian Harrison
Principal Administrative Officer
Hounslow Borough College

Hounslow Borough College is the only further education college in the London Borough of Hounslow so it was not surprising that the initiative for developing management information systems (MIS) came from the college itself rather than the LEA. This had distinct advantages in that it meant the analysis could be undertaken by the people who would make most use of the system rather than by those whose main concern would be the availability of the specific categories of information that the systems could produce.

On behalf of the college the LEA successfully applied in 1987 to the then Manpower Services Commission for funding for a research project which aimed to identify the MIS requirements of the college and to assess how best those requirements could be met.

The first task was to define aims and objectives — a crucial move in retrospect, because later on in the project reference back to the objectives served as a timely reminder of the need to judge alternative systems in the light of the perceived need for a fully integrated but flexible system. Among the objectives were the provision of cost efficiency indicators for the college and individual courses, the extension of the system to provide unit-costing information and the development of a system which would assist in the assignation of course priorities.

The project then fell into three parts. The first part consisted of a survey and assessment of existing management information systems within the college

and a study of the requirements of individual managers. The second part was an information gathering exercise on available MIS systems and the third part was an analysis of alternative systems, leading to a choice between them.

The initial survey included an assessment of existing systems and future management requirements, and a study of the data requirements of any new system.

The studies required a clear assessment of existing management structures and the varying degrees of delegation of responsibilities and levels of decision making. They involved an analysis of managers' perceptions of existing systems and the identification of gaps and weaknesses in those systems. There was then a detailed study of what management reports were currently produced, how they could be improved, what additional information was required and in what form. Finally the studies involved a survey of attitudes to MIS and an assessment of the different levels of understanding and commitment. This was a crucial aspect of the whole project since it revealed the extent to which one system might be more appropriate for our college than another.

ALTERNATIVE SYSTEMS

Alternative systems were then studied. It quickly became obvious that only two systems were sufficiently developed and offered the range of facilities and the back-up we required — FEMIS and Fretwell-Downing. Members of the project team made visits to see each system in operation. A visit to Gloucestershire College of Arts and Technology was made to investigate the implementation and operation of the FEMIS Student Record System; a similar visit was arranged to Berkshire County Council and one of its colleges to investigate the Fretwell-Downing Student Record System.

By this time the college was getting a good idea of what was available and what additional facilities were planned by the software producers. We were also conscious of some of the possible problems with each system.

The study of the requirements of our own college, combined with other information we had gathered, put us in a position to draw up a checklist of our requirements — taking account of what we knew was or would shortly be available and of the sort of external requirements which we gathered would eventually be placed on us following the Joint Efficiency Study. Both software suppliers were invited to present their wares at the college and to write to us stating the extent to which they could meet the requirements of

our checklist and at what price. Included in the checklist were hardware availability and training facilities, as well as software provision.

Both suppliers came to the college and performed. If I said one presentation was a slick, colourful hard sell and the other was a more laid back offering, anyone with experience of either firm could doubtless say which was which. There was a danger of being swung by the nature of the presentations rather than by the contents of the packages but we were not unduly influenced by the hard sell, largely because we were aware that the same approach had been adopted elsewhere and decisions had been taken without a full realisation of some of the possible drawbacks of the choice that was made.

On the other hand we also had reservations about a software producer whose marketing strategy seemed low-key to non-existent and had to ask ourselves whether, in a rapidly expanding market, such a firm would be able to provide the training and support services they promised and still meet their own targets for the release of future software modules.

To avoid making decisions based on impressions and suspicions we concluded the project by comparing each system against a number of criteria. These included costs, hardware availability and the facilities for and costs of training. We were then able to list the advantages and disadvantages of each system. We finally opted for FEMIS and have subsequently acquired and installed the FEMIS Student Record System on a Prime super-micro-computer. We have run the system live since September 1988 and so far have had no major problems, albeit that it has only been possible so far to use the system as a central administrative tool and the 'big bang' of opening it up to departmental use is yet to come. Nevertheless, we have expanded to eight live terminals and plan to install terminals in every department by summer 1989. We are confident that the system can and will become an effective management tool for departmental and central college managers and that the full benefits of the system will start to show during the next 12 months.

REASONS FOR CHOOSING FEMIS

We would not necessarily argue that our choice would be appropriate for all colleges. What FEMIS offered happened to suit our college's requirements and that was something we learnt from our analysis of our existing facilities, structures and attitudes as much as from our assessment of alternative systems. We also took into account technical considerations and long-term development potential.

Fretwell-Downing did appear to have some advantages over FEMIS, including an existing link between the student record system and a budgetary control module and, over all, the current availability of a wider range of modules as part of an integrated system, including an examination entries module.

On the other hand FEMIS came out better in terms of hardware capacity, expansion potential and a faster data entry facility. With FEMIS came ORACLE, a modern relational data-base which could be used to good effect for software other than FEMIS. The FEMIS system offered substantially greater flexibility for ad hoc queries and for user-designed reports. It also allowed for the creation of additional databases and user-operated screen design while also having more free text capacity. The ORACLE facility meant a very high level of file security which will be essential once terminals are spread all round the college — increasing the risk of unauthorised users trying to break into the system. Perhaps the most significant advantage of the FEMIS package, however, was the potential for linking FEMIS systems with other software including, for instance, a library loan system. Since installing the system we have been approached by another software company wishing to develop such a system on ORACLE in conjunction with the college.

Training, consultancy and the general availability of advice was another key area in shaping our decision to opt for FEMIS. It soon became clear that Fretwell-Downing has a more restrictive approach to who could and should be trained, how and where they should be trained and who could contact their support desk for help or advice. FEMIS, now EMIS Ltd., has a far more flexible approach. Training was available at the Further Education Staff College at moderate cost or could be supplied on-site at our college. Training packages could be assembled to meet the specific requirements of the college according to who was to be trained. The whole approach was more user-centred and less prescriptive than Fretwell-Downing and at a significantly lower cost. Our earlier research had shown that this sort of flexibility would be essential because of the different levels of understanding of MIS in different departments of the college.

I will not go into detail on the over-all cost comparisons between the FEMIS and Fretwell-Downing packages but we did analyse the costs, taking into account not only the initial setting up costs but also the anticipated ongoing costs of maintenance, licences, hardware and software expansion and training. If the cost of going for FEMIS had been approximately the

same as or even somewhat greater than the Fretwell-Downing option we would still have chosen FEMIS for all the reasons I have outlined above. In fact, we calculated that FEMIS would work out substantially less expensive, particularly in the medium to long term. That was clearly another influencing factor.

CONSTRAINTS

We were faced with certain constraints, quite apart from financial ones, relating to computerised systems already in operation at the college. We already ran the FEMIS core, timetabling and room utilisation modules on an RM-NIMBUS and the Fretwell-Downing budgetary control and order-processing systems on a Comart micro-computer. The former did not present any substantial problem as far as choosing the FEMIS student record module was concerned, but it has been necessary to adapt our coding system so that we have the same course codes on each system. The integration with the student record system is assured, with a trade-in-facility of existing software, as EMIS Ltd. is developing a new resource record module to replace the core and timetabling modules. I understand this should be available later in 1989 and will run on ORACLE alongside and linked to the student record module.

The financial control software we were running presented a problem in that FEMIS had no comparable system available or about to become available. Yet the Fretwell-Downing software was unsatisfactory for a number of reasons, not least because it ran on a single-user micro, so there was no direct access by other departments. At the same time the standard reports were over-complex, difficult to follow and it was not possible to amend the format to meet the 'user' requirements.

Furthermore Fretwell-Downing has recently given notice that it intends to stop maintaining this software in favour of its new multi-user version which is, I am told, a considerable improvement on version 7 which we operate. The conclusion drawn was that we would have to obtain a new commitment accounting system but if we wanted to standardise on FEMIS/ORACLE we could not opt for an upgraded Fretwell-Downing module.

We identified a possible solution before making our final choice in favour of FEMIS. ORACLE UK Ltd. was developing financial packages for polytechnics and colleges and both they and FEMIS told us they intended to develop a commitment accounting system linked to other FEMIS

software. In the meantime a commitment accounting system was being developed on ORACLE at North Staffordshire Polytechnic and our enquiries indicated that we could buy the system in as a temporary measure, if necessary, pending the development of financial software by FEMIS and/or ORACLE as part of a fully integrated management information system for colleges.

We have now acquired the software from North Staffordshire Polytechnic at a very reasonable price and are currently re-writing it to run on our Prime and expect to have it running live with effect from April 1989. It will be on the same system as the student record module and departments will soon be able to have access to both systems from their own terminals. This is a good illustration of the flexibility and development potential of the FEMIS/ ORACLE package. The FEMIS part is essential but having ORACLE as well opens up all sorts of extra possibilities.

One such attraction is the recruitment of staff. Being able to offer training and experience in the use and development of systems using an ORACLE relational database is a factor which has helped the college to recruit its computer manager and at least one of its computer technicians. This is not something that should be sneezed at for a college in west London's high-tech corridor alongside the A4, where computer technicians can potentially earn substantially higher sums in the private sector than in a local authority maintained FE College.

PROCESSES

I have sought in this paper to outline the processes we went through in selecting a management information system for our college. The processes did not end with the project because it is necessary to develop the systems we have installed as part of a continuous process to ensure that we get the maximum benefit from what the systems are able to offer. Nevertheless, we would not have got to where we are now and have succeeded so far with minimal disruption and with clarity about what we are trying to achieve had it not been for the efforts of the team who worked on the research project between 1987 and 1988. The team included Mike Moore, the vice principal, Roger Hickton who is a principal lecturer in the Business Studies Department, Howard Evans, a senior lecturer in the Science Department, Liz Slater, Hounslow's Assistant Education Officer for Further Education, and myself.

Copies of the project report are available, though again I stress that we do not pretend that it is any sort of blue-print. Rather it illustrates in detail the steps we took to evaluate our own and other available systems to aid us in choosing the most appropriate avenue of development to meet our own college's particular requirements. Nevertheless, I think the methodology we employed may commend itself to others who wish to undertake a similar exercise.

MIS: AN INTRODUCTION STRATEGY

David Atkinson
Staff Tutor, Further Education Staff College

Why do so many management information systems in colleges fail to have the impact expected of them? Why have some computer systems been consigned to dusty cupboards while others limp along in irrelevant obscurity? Why is there still a yawning gap between what college managers think they need in terms of information and what MIS actually provide? At a time when millions of pounds are being poured into college MIS, with more to come, it seems appropriate to look at a few factors wnich have caused failures — with a view to avoiding them.

Much has been written, particularly by the MIS suppliers, about implementation strategies. Such strategies concern themselves with hardware choice, software options, phasing, operating staff, data capture systems etc., and are a very necessary part of the acquisition of MIS. Some of the failures noted above can be traced back to poor implementation strategies. Is an implementation strategy, however, limited as it usually is to fairly technical matters, a sufficient condition for a successful system? This paper argues that another dimension, apart from the technical, should be considered — that dealing with the introduction of computer technology and quantitative data into college cultures. Alongside each implementation strategy we need an introduction strategy.

Little detailed study has been made of introduction strategies for college MIS. We depend to a large extent on anecdotal evidence, personal experience and the undeniable fact of system ineffectiveness. Problems associated with 'new office technology' have been analysed by the Work Research Unit (WRU, 1985) and this paper draws heavily on their work. Another study (Bennett, 1986) has examined the cultural area in FE colleges and

their response to MIS; this too has been extensively borrowed. Williams (1987) offers a sociological, but none-the-less useful, analysis of the same issues.

This paper sets out the two main problem areas which need to be tackled with an introduction strategy — staff resistance and management resistance. It goes on to suggest a simple format for dealing with these problems although it must be said that no amount of introduction strategy will successfully overcome deeply rooted resistance. In such cases one might be best advised to limit one's plans and wait until natural wastage has done its work. Perhaps the most useful strategy is simply to realise that an introduction strategy is needed. It is not simply a matter of unpacking the boxes, appointing a manager, and switching on.

Not all colleges will experience the resistances set out in this note and, in some, the degree of resistance may be negligible. Such colleges can afford to forget introduction strategies. But before you can jump to that happy conclusion you need to be sure that you are not yourself committed to MIS and therefore, like so many implementors, looking at the situation through rose-coloured glasses which distort the college's true situation.

STAFF RESISTANCE

The introduction of new technology into work situations where it has not hitherto been evident is often faced by staff resistance which can bring the system to a standstill. Such resistance can of course arise because of underlying organisational, managerial or personal tensions which crystalise around the information technology (IT) system. Where opposition to the system is merely a symptom of deeper tensions it is clearly necessary to defuse those before proceeding with IT. Where there are no such tensions, resistance may also be encountered because of opposition to the IT system per se; it can often be characterised as fear.

The key cause of staff resistance is probably fear of job loss, even where redundancy has not been mooted. Lecturing staff could well see (quite rightly) that MIS is intended to lead to greater lecturing efficiency and, in the absence of more students, consequent job losses. Fear will be particularly acute among those in 'inefficient' areas or those not central to training rather than education. The lecturer's response could be a refusal to co-operate with the system or even sabotage by inputting wrong data. The

fear would be compounded where lecturers feel the MIS has been imposed on them and is of no benefit to them.

Support staff could also fear job losses as many tasks traditionally undertaken by clerical officers are automated. Indeed, many MIS are justified on the grounds of saving clerical time, even though such savings look good on paper but rarely eventuate. It will be little comfort to the support staff to know that new jobs are being created to run the MIS, because existing staff are unlikely to get the new jobs. The value of long service and, therefore, the status of the long-established can be down played. Moreover, the creation of a new MIS section may be seen as a challenge to existing support staff structures — the chief admin officer may fear a loss of importance or a reorganisation leading to effective job loss.

Even where the fear of job loss can be overcome, there will be secondary fears — particularly among support staff. There may be a fear of skills loss because people value themselves and their jobs by the skills they have built up. To have an MIS come in and value these skills at nought is very stressful and will lead to resistance. The story has been told of registry office clerks who were highly skilled at dealing manually with enrolment forms and correcting errors in course codes, fees etc. When the MIS was introduced the system required the clerks to input all the data unchecked as the machine did the checking and output error listings. The clerks were appalled that their skills were no longer valued, that they were expected to input nonsense, and they became demotivated. Similar examples may be found in finance and timetabling. Not only is skills loss demotivating, it can also be linked with status loss and pay/prospects loss as less skilled workers can do what previously required a good deal of expertise.

Linked with skills loss there may be personal problems of fear of the unknown (i.e. the computer), and fear of being unable to adapt. Older workers, in particular, may not see the need to learn to live with the computer and one cannot but have some sympathy with them. It must not be assumed that everyone will welcome MIS with open arms or even with an open mind — they bring with them experience, preconceptions and personal inadequacies which can all result in fear and passive or active resistance.

COPING WITH RESISTANCE

Having identified some of the sources of fear, how can they be dealt with?

To meet the job loss fear management should explicitly address the redundancy issues, possibly by announcing that all efficiency readjustments will be handled through natural wastage. Failure to clarify these issues at the outset will lead to suspicion among the staff which will be difficult, if not impossible, to allay later. Openness about plans and intentions is therefore of the essence; openness, moreover, right from the beginning — not when the foundation has been laid and rumours have taken off. It is sometimes argued that such openness can give ammunition to a system's opponents and make them stronger. But how much stronger would they be if they could demonstrate by management secrecy that there was something to hide?

Proper system design should deal with deskilling fears. As part of implementation, every job affected by MIS should be reviewed with the postholder. Existing skills should be accommodated rather than replaced, and if there is a conflict the new system should merge with the old by changing the implementation strategy. With time, the strengths and benefits of the new should become apparent and then they will be adopted — but the pace should not be forced.

Fears of the unknown and being unable to adapt are essentially personal and should be dealt with on that basis. In addition to the job review, counselling should be offered to all staff expected to come into contact with the new system. A respectable bolt hole should be provided for those who, after counselling, cannot reconcile themselves to the new system.

As well as identifying and meeting such fears, a general antidote exists in the excitement and enthusiasm which can be engendered through training. Appropriate training should be organised for all, not just the select few who will operate the system. Extensive and lavish training programmes can repay their costs by warming-up the staff, by converting them to the cause of MIS. There is an inherent danger that staff will then cool down as the MIS fails to perform immediately, and this must be guarded against.

MANAGEMENT RESISTANCE

It might seem anomalous to suggest that management could introduce an MIS while still being opposed to it, but it can happen. For a start, management may not be monolithic; there may be competing groups within the top layer with some in favour of MIS and others against. Just because one

group triumphs in obtaining a system, this does not guarantee that the system will then be used properly. A college principal could agree to the acquisition of an MIS simply as a bargaining ploy in some larger affair.

The traditional college management processes have not emphasised a dependence on data or rationality. Cardy (1987) notes in the context of LEA/college budgeting that 'data are used in an impressionistic way to paint the background to negotiations'. He observes that 'negotiations are essentially a game play in which success is determined by factors such as negotiating skills and effectiveness in deploying or obfuscating relevant information'. In the absence of reliable data or agreed indicators it is hardly surprising that colleges have been managed in an incrementalist and haggling fashion.

The advent of MIS can change that culture. Bennett (1986) saw 'a great deal of uncertainty about shifts of power and influence in the institutions using the system'. . . 'the control of access to information and the selective retention or dissemination of information, and its timing are manifestations of power'. Once management information, as produced by an MIS, is in the public domain management decisions must respond to it. This effectively circumscribes management's freedom of action. The cultural change will be opposed by those who set great store by the traditions of bargaining and secrecy. It will also be opposed by those in the organisation who stand to lose power or influence, in just the same way as administrators who feared skills loss.

If, as Bennett notes, the MIS affects the locus of decision making — tending to centralise or decentralise — there may be organisational losers. The list of potential opponents seems endless. Can it be counteracted? If a college in its pre-MIS state is not already using some form of management data explicitly in its management decisions, it is bound to face problems. The simultaneous introduction of MIS and the resulting culture change can be too much. A common result is that the MIS remains an administrative tool unheeded by the management. MIS and management culture must be conceptually separated and their respective roles acknowledged. It is therefore a key part of an introduction strategy to secure management's informed consent to changing its practices.

MIS OWNERSHIP

At least two parallel hierarchies coexist in most colleges — the academic

and the administrative. Strategic management decisions on the future of courses or departments, on the deployment of resources, and on the choice of performance indicators, are usually located in the academic hierarchy. Why then is the management of MIS systems frequently located in the administrative hierarchy?

It might seem obvious that the MIS manager should be part of the administration, or even that the CAO should add it to her duties; after all, an MIS deals with enrolments, fees, finance etc., all of them firmly part of the administrative role. An MIS operating mainly in such a milieu is essentially an administrative system, not a true MIS. If it is to be of maximum benefit to managers the MIS must have a wider role. It must not be confined to administrative obscurity because it will not them be 'owned' by the whole college.

Ownership is a difficult concept, but one which repays study in any introduction strategy. Those who own the system see themselves as managing its priorities and having it serve their needs. In colleges where ownership has been vested in the administration, administrators will define its development. The consequence could well be that the academic staff find it of little use for their own needs and cease to pay it any attention. The final result is that the system degrades because errors remain unchecked by the academics, and administrators do not have the correct relevant data. Only if academics are admitted to ownership will they play their full part.

One solution to the academic/admin ownership problem is to create a hybrid systems manager role which serves both sides. Although she would probably be on a non-academic grade, she would be answerable to the vice principal (resources) in line management terms. As far as the CAO, registrar, and finance officer are concerned she would provide a data handling and analysis service, but would not duplicate their functions. There is a danger of such a person being frozen out by both academics and administrators, but it is a danger worth taking to maximise the relevance of the MIS to college management.

If such a solution is not appropriate, the issue of ownership and the potential problems should at least be addressed in the introduction strategy. The CAO, if she is to be the systems manager, will need to be involved in the college's strategic management processes to a greater extent than we currently see. There should also to be a clear distinction in the CAO's job specification between the traditional administrative tasks and the new cross-college MIS role.

EVALUATION

Colleges introducing MIS with ESG or similar funds will probably be obliged to evaluate their system for the benefit of the funders. This is the wrong reason for evaluation as far as the college is concerned, because it serves the outsider's ends rather than the college's. The purpose of evaluation is to get the best out of the system; setting up such an evaluation should be an integral part of any introduction strategy.

Analysts identify two basic forms of evaluation — the summative which delivers a verdict at the end of a process, and the responsive or formative which interacts through feedback with the process itself. A summative evaluation of the Titanic disaster would tell you why it happened; a formative evaluation might have kept the ship afloat. We are therefore mainly interested in formative evaluations.

Evaluation begins by setting up a group, preferably not coterminous with the MIS team, which is charged with drawing up an evaluation plan. This should start before implementation for two reasons: firstly, because much valuable formative evaluation can be fed back into the implementation and introduction strategies; secondly, because the group needs to get a reasonable idea of how college management functioned before the MIS if it is to draw any useful conclusions about the subsequent impact of MIS. Evaluation then runs in parallel with the MIS, indicating at regular intervals:

1 performance against pre-set targets, e.g. have certain reports been properly produced? have certain statistics been used? have deadlines been met?

2 qualitative assessments of the impact of the system, e.g. are management decisions being better made; does the college appear to be a more efficient and effective organisation?

The outputs of these evaluations are fed back at, say, quarterly intervals to the MIS managers with the intention of improving system relevance and performance. Evaluation, moreover, acts as a channel between the college as a whole and the MIS managers lodged in their ivory tower. After perhaps two years the work of the evaluation group can be handed over to an MIS user group; by then the system will either have reached a state of perfection or will be too entrenched to change.

THE FIRST STEP

The first months of MIS operation can be crucial to the way in which the college responds in the following years — give a dog a bad name . . . What you do in those first few months needs careful preparation. The first thing to accept and plan for is that the MIS will produce very little of real value for a considerable time. Colleges with MIS find they have a resources haemorrhage for up to 12 months, as considerable time must be spent inputting data before the system gives much in return. In setting up an MIS you are making an investment which needs time to mature. You must restrict initial expectations and not promise to produce major reports because you are sure to disappoint. The confidence of your users, once lost, will take a great deal of effort to regain.

Having dampened initial expectations, a good start can be gained by providing most system customers with something tangible at an early stage. Such lollipops can be fairly trivial but they serve to show that the MIS is up and running. Simple class lists, pre-printed envelopes addressed to employers, or even a staff list and telephone numbers (variously sorted), can all be used as the system's calling card.

It may be that the system has a negative reputation — 'it's meant to make us all work harder' etc. Counteract such negativity by making the system as user-friendly as possible. Much can be done through training but the atmosphere can be helped by giving the system a comic, perhaps ribald name. A competition among the staff might suggest such a name and could be a way of involving the college with MIS.

AN INTRODUCTION PLAN

Nothing can guarantee success when introducing an MIS, but there are ways of minimising the chances of failure. Here are a few points to note in any introduction plan.

1 Consultation on the planned MIS is probably the single most useful step that can be taken — consultation not only with the immediate users but with everyone who might feel threatened by the system or to whom it might be of some use. To be meaningful and therefore credible, consultation should start before deciding to install an MIS.

2 Decide the ownership issues.

3 Pay as much attention to training and counselling as you do to the technical specification.

4 Restrict the initial expectations of the MIS.

5 Provide lollipops.

6 Set up an evaluation group, separate from the implementers, to give responsive feedback. If you have encountered particular opposition during consultation, include the enemy among the evaluators.

REFERENCES

Bennett J and Lancaster D (1986) Management Information Systems in FE. **Journal of Further and Higher Education** Vol 10 No 3, pp 36-50

Cardy C and Crisp P (1987) Resourcing Accountability. **Educational Management and Administration** Vol 15 No 3, pp 171-181

Williams G (1987) using MIS to keep the college under review: information, power and contextual realities. **Coombe Lodge Report** Vol 19 No 8, pp 473-491

Work Research Unit (1985) New Office Technology. **WRU Occasional Paper** 34 (DES)

LIVING WITH A MANAGEMENT INFORMATION SYSTEM

Jean Skuse
CMS Co-ordinator and
Robert Case
Education Information Systems Officer
Education IT Section
Berkshire County Council

Berkshire began its implementation of a management information system for colleges in the summer of 1986. Until that time there was no systematic use of computer–assisted administrative tools within the authority, although one college had used the essentially AMS–based FEMIS.

The authority's specification for a system gave priority to a student enrolment–based system as the in-house mainframe software for supplying FESR was in need of replacement. It is not surprising, therefore, that the Fretwell-Downing system was chosen. It used the latest DRS 300 computers from ICL, a company with a long established relationship with the authority.

Now, more than two years later, is a useful time to review the systems evolution. The authority is currently reviewing its future development and decisions are expected shortly in this direction. The intention of this paper, however, is to reflect on the experience gained in a wider context and highlight some of the less publicised aspects of such a project.

HARDWARE AND SOFTWARE SUPPORT

At the start of the project the Education Department received considerable help from its own computer section, although it was not realised how much

work they had undertaken until their DRS 300 specialist resigned. Members of the Education Department had assumed that the support and maintenance licences paid to both ICL and Fretwell-Downing provided all the necessary back-up required. It soon became apparent, however, that there was an area of support not covered by either supplier.

All faults were initially routed through Fretwell-Downing's support desk because college staff were often unable to identify whether a problem was hardware or software related. Simple problems were resolved quickly while more difficult faults were sometimes referred back to the college as ICL's responsibility. ICL were then contacted and spent time trying to resolve the problem, only then to inform the college that the fault lay with Fretwell-Downing. The college had no knowledge or experience from which to question the validity of either of the support desks' responses. From User Group meetings it soon became apparent that several colleges were experiencing similar problems. This was resolved by the introduction of support log sheets identifying any problems with the system, copies of which are sent to the co-ordinator who pursues any problems which are not rectified within 24 hours.

When the initial problem arose there was insufficient time to train college staff in basic hardware awareness as it was only four weeks to the first full session of enrolments. Education staff undertook a three-day hardware and network training course to enable the colleges to route all hardware queries through the co-ordinator. Software faults were still routed directly to Fretwell-Downing.

Centralised hardware support made it easier to identify fault patterns. A few months later three colleges reported failures in back-up procedures and, when replacement modules did not resolve the problem, ICL were informed directly and given details of the sites involved. After a thorough investigation, working closely with one of the sites involved, it was discovered that there was a 'bug' in the archiving software supplied by ICL. Each college received, free of charge, a copy of the latest archiving software release and ICL also recommended the use of longer length tape cartridges.

Centralised support also ensured that Fretwell-Downing resolved faults rather than referring them as hardware related. It also highlighted problems when software 'bugs' occurred with update discs sent to each college. Following a particular software update, one college was unable to use its

system for over a week. To prevent this happening again new procedures were issued to the colleges. All software updates are now applied to the central system, held at Shire Hall, and tested before the colleges are told to proceed with the update. A back-up of the system prior to any application of updates ensures that if any unique problems occur they can be noted and the system retrieved to its original state, prior to updating. There is therefore no unnecessary down time at the college while the fault is rectified by Fretwell-Downing.

USER GROUPS

It was intended from the outset of the project to establish a user group within the authority. Working as a team, the user group would be able to effect Berkshire's corporate approach to CMIS. Regular monthly meetings enable the systems managers and the CMIS co-ordinator to join forces to identify problems, discuss new developments, exchange views and information and plan for the future. These meetings are usually informal affairs, although minutes are taken and action lists produced. This ensures that everyone has an opportunity to voice his or her opinion without feeling insecure or inadequate. An understated but important aspect of these monthly meetings has been the emergence of social evenings. Over the past two years five events have been organised and fully supported by the group.

The Southern User Group consists of representatives of local education authorities and colleges in the south of England who use the Fretwell-Downing software. The group started in July 1987 when it became apparent that there was more likelihood of a successful representation to Fretwell-Downing if the 11 member authorities spoke as one voice. The group now feels that it should extend its membership to include all LEAs in the south of England irrespective of the management information system in use.

Problems relating to the introduction of a computerised management information system (CMIS) will be common to all authorities, irrespective of the application software package installed. A forum which enables discussion and exchange of experiences among members of the Southern User Group can ony be viewed as a positive step forward.

It is likely that sub-groups will then be created to enable members using the same software applications package to discuss specific problems. Negotiations with the supplier to resolve these problems will, therefore, start from a stronger vantage.

TRAINING

Training has been a major issue for Berkshire. Fretwell-Downing's Training Centre is located in Sheffield and necessitates staff being absent from college for three or four days at a time, with travelling and accommodation costs adding an extra 90 per cent to the daily training rate.

An evaluation of the initial courses prompted the LEA to question the standard and quality of Fretwell-Downing's training provision. Copies of the evaluation reports were forwarded to Fretwell-Downing, although continuous monitoring of all subsequent training courses revealed little change in the situation.

The authority decided to set up its own training centre to enable the co-ordinator to provide localised training, which could be tailored to individual colleges requirements, at a greatly reduced cost. Colleges have made extensive use of these facilities during the past 12 months, not only in detailed application software training but also in workshop sessions. Considerable time has been spent in explaining how the software should be used and in giving college staff more confidence to use the system. The workshop sessions have given an opportunity for colleges to discuss the implementation of new modules and the possible administrative procedures required to integrate the system as smoothly as possible into the college.

Various suggestions can then be tested in conjunction with the software until an optimum solution is found. Procedure papers outlining the implementation plan to be adopted are then distributed to all college system managers. A similar approach is adopted for key reference files. This is an area in which Berkshire's positive, corporate approach to CMIS can be seen in practice.

However, basic training by Fretwell-Downing in each module used by the college has been a prerequisite of access to their support desk, a situation which has led to protest from the Southern User Group. Continuous pressure from the Southern User Group has led to a change in policy from Fretwell-Downing. Any member of a college may now telephone the support desk for help even if they have not been trained by Fretwell-Downing. However, if Fretwell-Downing perceive the problem as being a lack of knowledge of the system and a request for training over the telephone, the caller will be referred to their own systems manager in the first instance. No caller will be refused an answer but the college's systems

manager will be informed that training is required by this particular untrained member of staff.

Initial perceptions of the LEA's role in CMIS was one of co-ordination, with college staff having a greater knowledge and understanding of the system. This has not really happened and has resulted in many reports having to be written centrally and then installed in each college. The need to understand the basic data structures to create reports from the system has proved prohibitive for some members of staff who now feel unable to write their own reports to interrogate the database. Additional 'in-house' training has helped some staff but considering that the colleges have been using the system for two years, few have mastered Fretwell-Downing's report writing facilities.

A system that is so often dependent on one person is not consistent with the LEA's views of devolved management. A CMIS should give localised and fast access of management information to as wide a range of college managers as possible, who can then use it for improved decision making.

SUMMARY

The last two years have certainly been eventful and a great deal of experience and knowledge has been gained by everyone involved in the CMIS. Tackling problems head-on and resolving them quickly, accompanied by the establishment of procedures to prevent a re-occurrence, have proved productive. Regular review of procedures maintains a high degree of quality control.

In all, then, Berkshire's experience illustrates a number of characteristics which could be helpful for other authorities — whatever their chosen system. In summary these are:

- the benefits of having a CMS co-ordinator who can offer rapid responses to local problems and guide and develop the project
- the value of localised and thorough training
- the importance of a strong user group, both within the LEA and between LEAs, the latter being able to articulate powerfully the concerns of the users

- the need to clarify the relationships for support between software and hardware suppliers and

- the vital need for clear user procedures in running the system.

STAFF DEVELOPMENT AND TRAINING FOR MANAGEMENT INFORMATION SYSTEMS

Paul Crisp
Staff Tutor, Further Education Staff College

Any conversation on the subject of information systems[1] and their introduction and use in FE will soon get round to staff development. This is neither surprising nor unwelcome. The Department of Education and Science **User requirement report** (DES/WO, 1988) cautioned that 'a comprehensive training programme' would be required to support the introduction of management information systems. The Work Research Unit (1985) noted, however, that the introduction of information systems in organisations is not different in principle from any other major change. They urge that openness of communication about the changes and a participative style of consultation smooth the passage of change. Described thus, it can be difficult to distinguish staff consultation from staff development, particularly in the earlier stages of the process.

Colleges and LEAs appear to have taken these messages to heart, or so the number of commissions for consultancy and training received at FESC would suggest. It is, of course, possible that this enthusiasm for good preparation is loosely connected with a hint that bids to the DES for Education Support Grant (ESG) funds would stand a better chance of success if located within a clear strategy which includes staff development. The simultaneous availability of funds from the Training Agency (TA) to sponsor both the production of a staff development strategy and the staff development activity itself has inspired many to new heights of enthusiasm. It must, however, be said that staff development is not the universal

1 The terms Information System (IS) and Management Information System (MIS) are not synonymous but are often treated so. This paper used IS to describe any administrative/management system which makes use of data on computers. MIS refers to that particular kind of IS which handles the sub-set of **management** data.

a7

organisational sticking plaster it is sometimes treated as — and it can be very expensive. Spending priorities have to be assigned and some organisational ills and mismatches can only sensibly be resolved through more painful solutions such as staff, or even management, changes. However, it has been much more common to see expensive systems introduced with little by way of preparation, so the recent emphasis on and enthusiasm for staff development is a welcome antidote.

Accepting, then, that a programme of staff development and training is essential, what are the questions and issues to be dealt with? Put simply, they include:

- What's the difference, in this context, between staff development and staff training?
- Who needs which?
- What kinds are needed?
- When is it needed?
- Where do we get it?

STAFF DEVELOPMENT OR TRAINING?

I am conscious of the dangerous ground on which I am treading and have no wish to engage in the **education** versus **training** debate. In the context under discussion, by training I mean the transfer of particular, unproblematic skills applied in specific circumstances. A simple example would be the training of a typist or clerk to enter data using a specified screen (or set of them) for a particular Database Management Systems (DBMS). I would distinguish this from, say, a programme of seminars for administrative staff on the impact of information technology on their jobs. This I call staff development. It is of more general application and not primarily concerned with transferring specific skills; the purposes and likely outcomes are less clearly defined and are more likely to be described in terms of 'introducing' the subject or 'raising awareness' about it. Staff development may blend imperceptibly into other activities, such as staff consultation, and is much more often concerned with overcoming resistance to change than is staff training.

Identifying the training needs is, on the whole, a more straightforward exercise than designing a staff development programme. For a start, it is likely that one can quite easily identify the knowledge and skills required, but missing, which can most appropriately be supplied through some form of training. Typically these will be technical matters to do with, for instance, the functioning of the computer, its operating system, or the applications software[2] installed on it. Managers will know which staff require training for this knowledge or skill and staff will probably not argue about the nature of the need or how it should be met. None of these factors can be relied upon in relation to staff development. It is often said, for instance, that everyone in the organisation will be affected by the introduction of an MIS, so everyone needs some form of development. Senior managers cannot be certain, however, that all the staff will agree that they have any staff development needs, and they may be inclined to dispute the management's interpretation of what they are and how they might best be met.

WHO NEEDS WHICH?

The involvement of staff with information systems can be classified into two broad types. Those who manage information systems (or policy about them) and the flows of information processed by them are one; we can call them the **providers**. The other group contains those people who make use of the system — the **users**. These are rarely mutually exclusive. It is very probable that the people providing source data (on enrolments, say, or staff timetables) are users of the system's outputs in a variety of ways. Nevertheless, the distinction is useful because the development and training needs of each group will be very different.

THE PROVIDERS

Who are the 'providers'? This will depend very much on the precise set of circumstances. What point in the development process has the college or LEA reached? How big and/or complex is the college or system? What are the resources available to support it? All these factors will influence the kinds of roles to be performed and the numbers of people performing them.

2 Applications software — the set of programs which provide the functions the end-user wants; as distinct from things such as the operating system software, necessary but obscure parts of the supporting system.

Some of the roles may be combined in one person, particularly in smaller systems, who will often be called some variant of MIS officer. Nevertheless, at some point the following jobs will need to be done.

The policy maker — this may be one person but it is more often a group of people at LEA and/or college level responsible for taking broad strategic decisions about information systems. They are at their most active before any computer-based system is acquired but IS policy has to be reviewed and re-made as circumstances change. This group often includes members with particular expertise in computer-related subjects and others who are knowledgeable about the workings of the college or FE system. The training needs of such a group are likely to be different for each individual although they may benefit collectively by working on their ability to function as a group.

The system manager — I use this term to describe the individual with overall responsibility for integrating the information system with the rest of the management system of the institution (or FE service at LEA level). The crucial point is that the information system is just one of the systems being managed. This task is often seen as administrative and given to the Chief Administrative Officer. Elsewhere it is seen as an appropriate task for a vice-principal. The skills needed by this person are more to do with tasks of management than knowledge of computers. However, s/he does need to understand the capabilities of the technology and the nature of the interaction between the institution's management system, and the people who service it.

The system administrator — is responsible for managing the information system on a day-to-day basis. This function is often confused with that of the system manager but is a more self-contained set of tasks associated with managing the people running the system, managing the flow of 'raw' data in and reports out of the IS, designing data capture documents, producing coding structures, organising the personnel responsible for the administrative system in which the computer system lives, and so on. In a commercial computing environment, these tasks include those which would typically be called 'data processing operations'. They are not radically different from the routine administration of a college or LEA with which most administrators are wholly at home. In relation to computers, however, further education has been rather naive about the elaborate procedures required to maintain the big, complex ISs now being introduced to colleges. Training can help the translation.

The computer manager — as the name suggests, runs the computer and the processes which support it. S/he has to understand the hardware, its (their) operating system(s) and the applications software in use. This is a task of increasing complexity as colleges acquire mini-computer or super-micro-based information systems and integrate them (or try to) with all kinds of computing equipment and software. This individual's training needs are likely to be seen primarily as the technological aspects of the system(s).

The computer programmer — is most likely to be the same person as the computer manager, at least in the smaller installations or in the earlier stages. Some institutions will decide that they have no intention of 'programming' anything and will dispense with this function. Others will recognise that they need someone who can produce applications software or modify proprietary software to suit local circumstances. The programmer's training needs are mainly technological.

The data input clerk — is responsible for getting information into the system, usually by typing it in at a terminal. Where this is a self-contained job, defining the training needs is fairly straightforward. Where the institution has chosen to decentralise the data input task (to departmental or faculty offices, for instance), defining the training needs gets more complex as, in these circumstances, clerks are likely to be called upon to work relatively unsupervised, have a good grasp of the college's systems and procedures and understand how the applications software works.

THE USERS

All of the people identified as 'providers' are also 'users' at some point or points in the chain from data capture to report production and application. This is particularly true in organisations which have combined the functions described above with each other or with other roles. The big operational systems now being introduced into colleges (as described in the **DES User requirement report**) will eventually connect with every aspect of college life. The Education Reform Act and improvements in marketing techniques are just two of the factors giving added impetus to the trend towards needing more information about the college's activities. Even those systems strictly defined as management information systems are handling data which are (or should be) relevant to everybody from course teams 'upward'. In the very largest sense everybody in the organisation is a 'user', albeit, for some, at third or fourth hand.

The problem of defining users' staff development needs is a considerable one. The MANDIT project (Fonda, 1986) looked at the implementation and use of information technology in business and the management development required to support it. They concluded that a number of problems can flow from badly managed design and use of IT. They identified some technological difficulties but noted that '. . . they are results of the ways in which the choice, design and implementation of information technology have been **managed**' (their emphasis). MANDIT suggests a number of 'management competencies for information technology'. These are summarised in the Appendix to this paper but the ones of relevance here are:

- ability to work with management concepts (e.g. inputs/outputs, priorities, performance criteria etc.)
- understanding organisational systems
- knowing how to find out
- knowing what it is not necessary to know.

Few of these have much to do with the technology *per se*. The introduction of information systems often brings to the surface previously unacknowledged needs for staff development in relation to more general management skills. For most users, information systems are tools of the management process and staff development activities are probably best focused on the management processes which use the tools.

The other set of users — college personnel in general — are often perceived, or at least represented, as being primarily the means by which data can be offered up to the system. They are therefore a 'problem'; specifically they are a problem of data accuracy and promptness. Many proposals for 'staff development' are little more than a plea for a foolproof way of getting the 'troops' to do as they are told. Here the exhortations of the WRU are relevant. They suggest that 'people . . . who will use or be affected by new technology should . . . be involved from the outset in the design and implementation of the system.' (WRU, 1985) They noted that this not only improved system design, it 'gain[s] the commitment of those affected. Such involvement undoubtedly helps the smooth running of the new system, and is essential as preparation for future change.' Manipula-

tion, thinly disguised as staff development, will not substitute for the proper participation of the 'end-users' in the development process.

WHAT KINDS OF STAFF DEVELOPMENT/TRAINING ARE NEEDED?

We have begun to address this question in the preceding section. The staff development/training requirements will clearly be different for different functions and will change over time. The degree to which some of the functions set out above are combined with others will also affect the staff development/training needs. Nor am I suggesting that development needs can be wholly analysed in terms of roles; some will be to do with the characteristics of the individuals in the jobs. We can perhaps suggest some of the roles and functions which can be assisted through staff development/ training.

Information systems strategy — This is largely about assessing the need and possible (emphasis on 'possible') ways of fulfilling it. It includes identifying the information requirements (sometimes known as the 'user requirement' or 'system specification'). This might be done in-house, or through a consultant, or not at all. In most cases the process includes some form of systems analysis and, even if the analytical task is undertaken elsewhere, someone has to guide the process. The next step is evaluating alternative ways of fulfilling the specification. Colleges and LEAs are increasingly abandoning the idea of producing a home grown information system and are, instead, comparing between and selecting from the few commercially available systems. Strategic skills in this territory include the ability to manoeuvre round the prejudices of the many interest groups (including the computer unit, local authority treasurers, politicians, the principal and the consultant you employed). This is, in itself, an arcane art. The continuing task is ensuring that IS strategy remains the servant of overall policy at college and/or LEA level.

Implementation strategy — Identifying the system requirements and how to satisfy them is only the beginning. If the development is to be successful, some work has to be done on the processes of implementation. The particular elements of the implementation strategy will depend very much on the context. In most circumstances, however, we can expect that the elements will include the following.

— Assessing the appropriate phasing of implementation in the light of prevailing political and financial conditions. The present 'political' conditions are largely those of the LEA. With financial delegation, however, the locus will shift to the governing body. Also to be considered at the moment are the requirements of the DES and the TA as they each control access to resources in the form of ESG and the Mutual Development Fund.

— Establishing a timetable which is technically feasible but which is also alive to the realities of the organisational culture into which it is being introduced.

— Identifying key 'gatekeepers' and potential blockages. Gatekeepers are those individuals whose co-operation is important to the success of the implementation. In the early stages of development these are typically senior academic managers but, in the implementation stage proper, they tend to be individuals positioned at nodal points in the organisation's communications systems. The gatekeepers are often administrative, clerical and/or secretarial staff. Dealing with the resistance of individuals and groups requires personal and inter-personal skills of a high order. It is important to remember that the rationale for the change, which the enthusiast finds wholly persuasive, may have a negative impact on members of staff who feel threatened by it and who can see no substantive benefit for themselves (or, by extension, for the organisation). Other forms of blockage might be accommodation problems (where do we put the equipment?), difficulties with suppliers (for instance, provision of internal cabling), finance and cash flow problems (getting the work authorised and paid for in the right financial year) and so on. Overcoming these blockages is all part of the task of project management. (A useful guide to the tasks of project management can be found in **How to manage a project** produced by the PICKUP Unit (Adult Training Promotions Unit) of the DES.)

— Organising the staff supporting the system. This might include appointing new staff (and, therefore, assessing the need, producing job descriptions, organising advertisements, selection procedures etc.) but will certainly involve changes in the

functions and responsibilities of existing staff. A major part of implementation strategy is dealing with the fears and concerns of staff, often through their trade union representatives.

Awareness — Some of the fears mentioned above can be allayed if staff have a good understanding of the purpose and operation of the new systems being introduced. As important as real understanding is the feeling by staff that management is genuinely interested in their views and is putting real effort into discovering them. I noted earlier that 'awareness' raising is often a euphemism for persuasion or consultation. Implementers should be careful to understand the purpose of their awareness raising sessions as some objectives will be compromised by others. It is very difficult to eulogise on the benefits of an MIS and simultaneously explore the practical problems of implementation to be overcome.

Technical: hardware and software — This is usually self-evident. If new hardware is being acquired, training in its operation will be essential. Similarly, the installation of new systems or applications software will necessitate the training or re-training of appropriate personnel. There are some problematic issues however. For instance, should the organisation expect to have hardware maintenance expertise or is this to be handled by the supplier through maintenance agreements? How many people should receive formal training in the operation of applications software? The cascade model seems the most cost-effective but leaves the organisation vulnerable if a key individual moves to another job or is not an effective communicator.

Administration — This can be the most difficult part of the implementation process. Student record and financial systems, in particular, operate in the heartland of institutional administration. Existing procedures and processes are often deeply embedded in the collective psyche of 'the office' and change is often resisted both overtly and passively. There are many other hard issues to be resolved. For example:

- preparing detailed user procedures for input handling, documentation, input forms, batch and work-in-progress control etc.;
- establishment of coding systems and assembly of reference files;

— liaison with other systems (e.g. LEA and local authority treasurer);

— identification and design/customisation of outputs, standard reports etc.;

— arrangements for pilot or parallel running and subsequent going 'live'.

Operations — The job of getting the data into the system and updating the database is fundamental to its success. This is still primarily a keyboarding task undertaken by a data preparation operator who is either a specialist or does 'data prep' as part of a wider clerical/admin role. Either way, the operator will need a detailed familiarity with input procedures, data input screens and paper handling procedures such as batch processing, validation and correction routines. As I noted earlier, the issue of **who** gets what kind of training in this area depends a lot on how centralised and specialised the data input function is in the college.

Use of outputs — This is ultimately what it's all for. We can be fairly confident that **some** use will be made of MIS outputs. Experience suggests that they are used extensively in the day-to-day administration of institutions. The problem in the past has been to persuade managers that MIS has something to offer them. Staff development is clearly one of the approaches to be used in promoting the management use of MIS data, although the current tide of events may prove to be a more effective encouragement of this use. Most staff development activity is (or ought to be) related to the management tasks to which information from the MIS might be applied. In general, staff development should be built around the management role. Arguably, however, one of the most effective ways of promoting the use of the system is to make it as accessible as possible. This implies the provision of very many terminals and that all or the majority of managers in the institution should be trained in the use of the MIS's report writer and/or enquiry language. A further consequence of this direct-access approach is that managers would need to be or to become very familiar with the structure of the data base and the data contained in it. This has many advantages for the quality and use of the data but is expensive to provide. At the very least, however, users should have ready access to a catalogue of the standard management reports available to them together with some explanation of what each report means.

WHEN IS IT NEEDED?

If we were considering a single, large-scale computerisation project, it would be relatively straightforward to devise a schedule of staff development/training which attempted to make the training available at just the right time. The sequence would probably look something like the following.

- **Pre-development phase** — up to and including the decision to develop or acquire a particular MIS solution. The phase would include most of the work on information systems strategy and probably some of the elements of the implementation strategy. It is possible that some awareness raising sessions would be mounted at this stage.

- **Pre-installation phase** — encompasses the period up to and including the delivery and installation of the hardware and software. Much of the technical training of specialists will occur in this phase. Training and development relating to the administrative systems has to be accomplished here. Awareness/familiarisation sessions may either start or continue in this phase. Operator training might be started here although this should depend on how long the phase is expected to last. It is generally thought bad practice to provide skills training more than a few weeks before the trainees can practice and develop these skills for real.

- **Post-installation phase** — includes the commissioning of the equipment, any pilot or parallel running, finalising data structures and administrative systems, setting up reference files etc. A minimum number of operators must be trained by this point. Some initial definition of output reports will typically be happening during this phase.

- **Operational phase** — begins when the system goes live and continues from then on. Little training is likely to happen initially as everything is time-critical and no one can spare the time or attention to organise it. There might be some operator training carried out as the system goes live and more operators are required. Otherwise, most 'training' is of the on-the-job variety. Later, development and training needs are likely to

focus on refining support systems (e.g. data flow control) in the light of experience, updates and revisions to the software, hardware maintenance etc.

— **Reports phase** — is the latter part of the operational phase when data is in the system and available for interrogation. Staff development needs may include both the techniques for extracting data from the MIS and approaches to encouraging managers to ask for and make use of the data. These may involve training managers in the use of direct data extraction techniques (i.e. using the report writer and/or query language).

Life is seldom as tidy as indicated in the sequence outlined here, unfortunately, and there are many practical complications to deal with. The first is that MIS development is rarely on a *tabula rasa*. The project is usually a series of developments, some of which overlap. An institution may simultaneously be in two or more of the phases suggested above. Second, it is common for the acquisition decision to be taken late and/or for the target implementation date to be early. As a result, the various phases are compressed into far too short a period, one or more of them is left out and work on a particular part of the project (for instance, designing a course coding system) started, even completed, before any training takes place. Third, the appropriate personnel are frequently not in post in time for staff development and training to happen at the optimum time. Staff resistance, individual or collective, may bring about similar delays. Fourth, it may not be possible to identify an appropriate training provider or secure places on the relevant courses at the time they are needed. Finally, but this is not an exhaustive list, there may be insufficient funds to pay for the staff development and training needed.

In theory, most of the problems mentioned above ought to be good cause to postpone the launch date. But, for political and other reasons, this is not often a realistic option. The organiser of the staff development/training therefore has little choice but to proceed with absolute priorities. These are likely to be:

— Training at least one person in the detailed operation of the hardware and the operating system.

— Training at least one person in the operation of the applications software.

— Training a core group (maybe only one) of operators.

— Getting guidance from the software suppliers on the essential/desirable data administration procedures.

SOURCES OF TRAINING

Some kinds of training can be obtained only from particular sources. Others might be available from a variety of organisations or individuals. Some staff development can only realistically be undertaken in-house or on-the-job.

You are unlikely to be able to get training on the specifics of the applications software from any source other than the software vendor. This is less true if you are using a generic microcomputer package (e.g. dBASE III), as many training providers offer courses in these products. They are, however, unlikely to be tailored to your specific needs. It is, in any case, improbable that the kinds of major systems envisaged in colleges would be built in packages of this type. The same considerations apply to hardware training. Vendor-provided training is typically the most expensive form with courses costing between £200 and £500 per **trainee** day (exclusive of accommodation costs). If you have a reasonable number of people to train (operators, for instance) it is often better value to arrange for the vendor to provide on-site training. This is likely to cost around £1000 — £1500 per **trainer** day (plus expenses) and you can usually negotiate a customised training programme. The larger proprietary MIS commonly have training 'credits' included in the purchase price.

There might be scope for satisfying some training needs through third-party providers. Many agencies which offer word-processing training (including local FE colleges or, even, a department of your own!) might be able to organise some operator training. This is more likely to be a viable option when using common commercial products. This is likely to be significantly cheaper than vendor training but the trade-off might be that it would be less specific to your needs.

A lot of staff development can not be done by outsiders. Awareness training, for example, requires a profound understanding of local circumstances, organisational culture etc. Similarly, much of the training on data administration must be internal, otherwise you would spend all

your time briefing the external trainer on the detail of your procedures. Help can be purchased in the design and delivery of staff development of this type. FESC, for instance, offers a consultancy service in this area and software vendors can also assist. You might find that the latter will do awareness raising for you as a marketing exercise for their products. Be careful that this approach does not raise awareness in undesirable ways.

Much of the on-going training will, of necessity, be on-the-job. Training in local operating procedures — for example, handling input forms — will be on this basis. Unless staff turn-over is very low, many institutions will find it difficult to send all operators (say) on external training courses, particularly the comparatively expensive ones run by software and hardware vendors. Most organisations will expect to adopt the 'cascade' model in which a few staff are sent on external courses and they, in turn, train others within the institution. The is a well-tried and reasonably effective way of meeting some training needs **providing** that:

a) The people selected to do the 'cascading' are competent trainers, rather than (or in addition to) people who themselves need the training.

b) The 'cascaders' know in advance that they will be called upon to undertake this work.

c) It is adequately resourced. Cascade training is undoubtedly cheaper but it still costs money and, more importantly, the time of both the trainer and the trainee. It must be organised, timely and systematic if it is to be effective.

AND FINALLY . . .

It is a truism that no organisation can ever afford to meet all the staff development and training needs which can be identified. In general, the training provided by software and hardware vendors is the most expensive and on-the-job training the cheapest. Some external training is absolutely necessary; some is not worth having. On the other hand, trying to provide all training internally can prove to be counter-productive. Training, and the willingness of employers to provide it, can be seen by staff as a sign of how important they are in the eyes of the management. Staff development and

training have important roles to play in encouraging co-operation with and ownership of the new systems. Sending more staff than absolutely necessary on expensive external courses is one way of showing you care!

Lastly, this paper has concentrated primarily on the staff development and training needs likely to emerge during the development phases. It has also focused implicity on the more traditional forms of staff development/ training — courses of one kind or another. Staff development, however, takes many more varied forms that this and can include a range of more interesting techniques such as job shadowing, job exchanges, work-based projects and assignments and so on. A discussion of the place of these approaches is, however, a subject for another paper.

REFERENCES

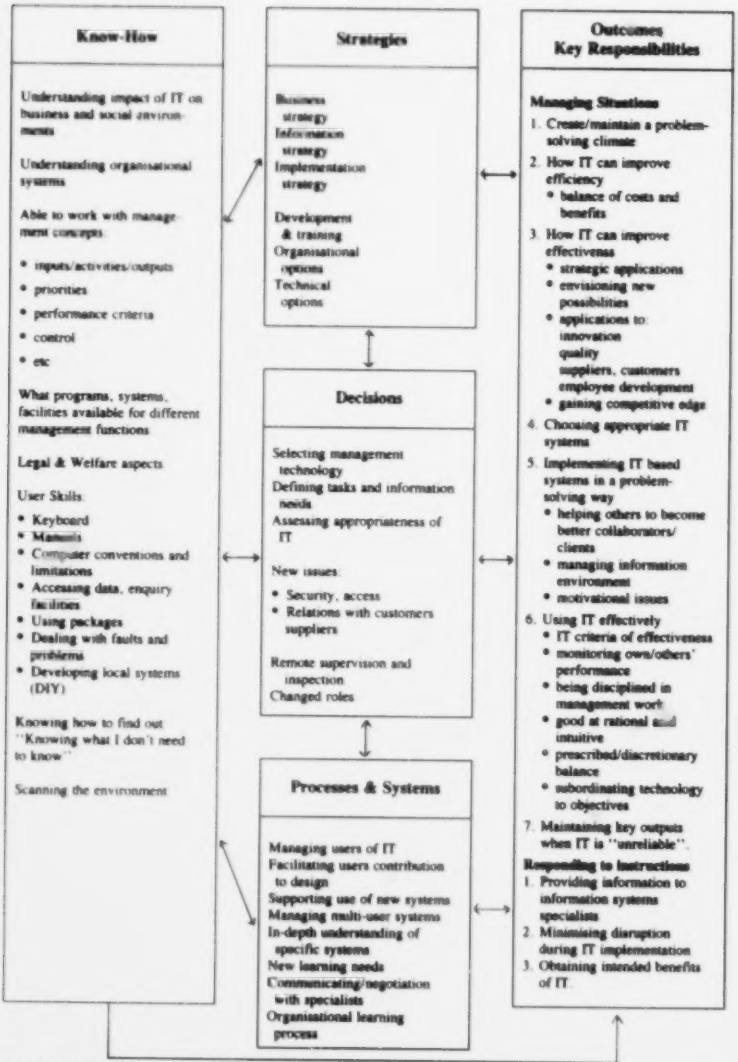
Education & Science, Department of & Welsh Office (1988) **User requirement report for computerised management information systems for further education colleges: a report of a study funded by the Department of Education and Science.** DES/WO

Fonda, Nickie (1986) **MANDIT: Management Development for Information Technology. Part 2: Aids to Development — Unit 8: Planning management development for information technology. A guide for management development managers.** Manpower Services Commission

Work Research Unit (1985) New Office Technology. **WRU Occasional Paper 34** (DES)

APPENDIX

Management Competencies for Information Technology


Know-How
Understanding impact of IT on business and social environments
Understanding organisational systems
Able to work with management concepts:
• inputs/activities/outputs
• priorities
• performance criteria
• control
• etc
What programs, systems, facilities available for different management functions
Legal & Welfare aspects
User Skills:
• Keyboard
• Manuals
• Computer conventions and limitations
• Accessing data, enquiry facilities
• Using packages
• Dealing with faults and problems
• Developing local systems (DIY)
Knowing how to find out
"Knowing what I don't need to know"
Scanning the environment
Strategies
Business strategy
Information strategy
Implementation strategy
Development & training
Organisational options
Technical options
Decisions
Selecting management technology
Defining tasks and information needs
Assessing appropriateness of IT
New issues:
• Security, access
• Relations with customers suppliers
Remote supervision and inspection
Changed roles
Processes & Systems
Managing users of IT
Facilitating users contribution to design
Supporting use of new systems
Managing multi-user systems
In-depth understanding of specific systems
New learning needs
Communicating/negotiation with specialists
Organisational learning process
Outcomes
Key Responsibilities
Managing Situations
1. Create/maintain a problem-solving climate
2. How IT can improve efficiency
• balance of costs and benefits
3. How IT can improve effectiveness
• strategic applications
• envisioning new possibilities
• applications to: innovation quality suppliers, customers employee development
• gaining competitive edge
4. Choosing appropriate IT systems
5. Implementing IT based systems in a problem-solving way
• helping others to become better collaborators/clients
• managing information environment
• motivational issues
6. Using IT effectively
• IT criteria of effectiveness
• monitoring own/others' performance
• being disciplined in management work
• good at rational and intuitive
• prescribed/discretionary balance
• subordinating technology to objectives
7. Maintaining key outputs when IT is "unreliable".
Responding to Instructions
1. Providing information to information systems specialists
2. Minimising disruption during IT implementation
3. Obtaining intended benefits of IT.

From MANDIT: Management Development for information technology. Part 2: Aids to Development — Unit 8: Planning management development for information technology. A guide for management development managers. Reproduced with the permission of the Controller of Her Majesty's Stationery Office.

THE FESC GUIDE TO COMPUTER JARGON

(The Not-Completely-Serious Edition)

Paul Crisp, Staff Tutor
Mick Fletcher, Staff Tutor
Dave Senior, Formerly Computer Manager

Further Education Staff College

One of the problems confronting ordinary mortals trying to get to grips with computing is the words used. There are several difficulties. The terms are sometimes technical ones exported into everyday life. Many are American in origin. The worst are those which are an eccentric usage of an otherwise perfectly ordinary English word or expression. Finally, there are several words for what seems to be the same thing.

The following pages list many of the jargon terms we have encountered on workshops and other events at the Staff College. This Guide is, of course, not all-inclusive. We have concentrated on those terms which relate to micro-computers and the software used on them but, as mini-computers are now appearing in education, we have tried to pick up bits of jargon associated with them as well. Nor are the entries in the Guide technical definitions. Computer experts might well quibble with the descriptions offered here. We make no apologies for 'inaccuracies' — our purpose is to try to make the opaque a little more transparent and we have been happy to (over-)simplify to achieve this end. Finally, a warning: we have not been overly serious about the endeavour. It is probably best **not** to quote from this Guide in the presence of two or more programmers — we take no responsibility for the consequences!

The following conventions have been followed in the Guide:

- Entries are in alphabetical order.
- Each entry header is in CAPITALS and **bold**.
- Cross references in the text to other entries are in **bold**.

4GE

Fourth Generation Environment. (cf **generations**)

4GL

Fourth Generation Language. (cf **generations**).

ABORT

To terminate a **program** or other function while it is still running. (cf **crash, control-c**)

ADAPTOR

See **monitor**.

ADDRESS

The location of a piece of information in the computer's **memory**. Much of the flexibility and power of **programs** is achieved by referring to the address in the program, rather than the particular **data** contained there. In this way the program can carry out operations on different data without having to rewrite the program. Also refers to the position on the **screen** (as in screen address). Sometimes used in **spreadsheets** as an alternative way of describing **cell references**.

AI

An abbreviation of **artificial intelligence**.

ALGOL

A **high-level programming language**. Stands for **ALG**Orithmic **L**anguage.

ALGORITHM

A formal definition of a solution to a problem. At the simplest level an algorithm is a mathematical equation. Thus the algorithm for finding the area of a rectangle is the product of the length of two adjacent sides (L x W).

ALPHANUMERIC

Strictly the numbers 0–9 and the characters A–Z (both upper and lower case). Often extended to include all the 'printable' characters on the computer **keyboard**. (cf **ASCII characters**)

ANALOG(UE)

The representation of **values** as continuously variable (typically a variable voltage). Some early computers were **analog**. Contrasts with **digital**.

APPLE

A computer manufacturer. Produces the Mackintosh (known as the 'Mac') series of computers. Famous for first using the **WIMP** type of user **interface**.

APPLICATION

The program (or usually group of programs) designed to carry out the functions required by the **end-user**. Thus a **word-processing** application or a **spreadsheet** application. Contrasts with **operating system**.

APPLICATIONS GENERATOR

A feature of the **fourth generation environment**. Allows programmers (and non-programmers) to create **applications** quickly, relatively easily and **interactively**.

ARTIFICIAL INTELLIGENCE

A branch of computer science concerned with mimicking 'intelligence' with computers. A particular feature of artificial intelligence systems is their ability to learn. (cf **expert systems**)

ASCII CHARACTERS

ASCII stands for American Standard Code for Information Interchange. It defines standard codes for all the 'printable' characters on the **keyboard** and some others (a total of 128). The use of this standard theoretically helps program designers to produce one program which will run on lots of different computers.

ASSEMBLY LANGUAGE (or ASSEMBLER)

A programming **language** more understandable than **machine code** but less like ordinary English than a **high-level language**. Still pretty impenetrable to ordinary human beings.

AUTOEXEC.BAT

A special form of **batch file** which the computer automatically **executes** when it is first switched on or is **re-booted**.

BACKGROUND

A computer operation given lower priority than one in the foreground. Thus 'printing in background' sets the computer to **print out** a document while running another program. The printing will happen slowly because the computer can only do it in the gaps between processing the foreground task (for instance while you are typing the next **instruction** at the **keyboard**).

BACKUP

Making a **copy** (onto, for instance, **floppy disk**) of important data for security purposes. Usually forgotten until after something dreadful has happened.

BADLY BEHAVED

A term used to describe **applications software** which issues **instructions** direct to the **hardware** and bypasses the **operating system**. 'Badly behaved' because the software will only work reliably with a particular machine (or group of machines). On other machines it will do unpredictable things — like nothing at all, or filling the screen with

gobbledygook, or causing the computer to **hang** or **lock**. Very prevalent with **games software** but pretty common with any software which does clever things with the **screen display**. A strong reason for following the adage 'try before you buy'. (cf **port–2, well behaved**)

BASIC

A particular and very popular **high level language**, mainly for micro-computers, originally simple and easy to learn. The acronym is supposed to stand for Beginners All-purpose Symbolic Instruction Code.

BATCH

A method of running a **program** in which the information is collected together and then submitted for processing in one go (batch processing). Contrasts with **interactive**. A common example in FE is the handling of enrolment data, where the information (usually enrolment forms) is sent in bundles to the LEA computer centre. Also used as a noun to describe the bundle of forms (a batch).

BATCH FILE

A **file** containing a number of (**micro-computer**) **operating system commands**. Typically used to **execute** a commonly used series of instructions to save having to type them in over and over again. (cf **autoexec.bat**)

BAUD (RATE)

The rate at which a computer sends data to another **device**, for example, a printer or another computer. It is usually the same as '**bits** per second'. Printers typically work at 1200 or 2400 baud.

BINARY

Most computers now work on the binary principle, that is a calculating system wholly based on 'on' and 'off', usually represented as 1 and 0. These are the **binary digits**, hence 'digital computers'. All **programs** have eventually to be translated into binary digits, the only **language** the computer can actually work on. (cf **machine code**)

BINARY DIGIT

Commonly abbreviated to **bit**. (cf **binary**)

BIT

Abbreviation of **binary digit**.

BOARD

Misspelling of word describing most **end-users**' reaction to computer manuals. Also word for a sheet of material containing a connected group of computer components. (cf **card**)

BOOT(STRAP)

Can be used to mean starting up the computer. Slightly more specifically it refers to the process of **loading** the **operating system** into the computer (hence 'booting the system'), a necessary stage before the computer can do anything else. (cf **re-booting**)

BUFFER

An area of **memory** for temporarily storing information in transit from one place or **device** to another (e.g. from a computer to a **printer**). Used to compensate for the different operating speeds of each device, thus allowing the computer to get on with another task without having to wait for the printer. Also, things at the ends of railway lines.

BUG

A term used to describe any kind of defect. Also used by **software houses** to describe the more persistent complaining customers. Corrections for bugs are known as **fixes** or **patches**. These you usually get for free. Big ones, or large collections of them, are called **upgrades** or new **versions**. These you pay for. (cf **end-users**)

BUNDLE

All the bits of **hardware** and **software** sold as one package. Commonly used to describe the software provided 'free' with computer hardware. But remember — there ain't no such thing as a free lunch!

BYTE

Eight **bits**. Typically represents one character of information. (cf **ASCII characters**)

CA (D/M/E/L)

Short for computer aided anything. The examples are Design, Manufacture, Engineering and Learning.

CARD

Same as **board**. Often used as shorthand for an extra bit of circuitry which can be inserted inside a computer to extend its facilities (e.g. a **monitor adaptor** card).

CARRIAGE RETURN

Used, as in a typewriter, to terminate the end of a line (on the **screen** image or on printed **output**) and start at the beginning of the next one. Produced by pressing the **Return** or **Enter** key on the keyboard. Also used to instruct the computer to **execute** the **command** typed in at the keyboard.

CELL REFERENCE

Used particularly in **spreadsheets** to describe the location of a **value** by its horizontal and vertical co-ordinates (e.g. A5 means in the first (A) column and the 5th row).

CENTRAL PROCESSING UNIT (abbr. CPU)

The bit of the computer which actually **executes** the instructions and does the calculations. Sometimes used to describe the cubiform box which contains the CPU (but also contains a lot of other stuff, like **disk drives**).

CHIP

An abbreviation of 'silicon chip' more-or-less synonymous with **integrated circuit**.

COBOL

A **high level language**. Acronym stands for **CO**mmon **B**usiness **O**riented **L**anguage.

CODE

Tends to be used by programmers to mean any kind of **program**. Some programs are written by the programmer as a series of **instructions** in words very like ordinary English. These are then converted by another program (called a **compiler** or **interpreter**) into the **machine code** (sometimes called micro code) which the computer can actually understand. Code and program are therefore more-or-less interchangeable terms.

COFFEE

Liquid used to fuel programmers. If ingested by computers, causes them to **crash.**

COFFEE BREAK

Any scheduled or unscheduled pause in computer operations. (cf **crash**)

COLD PRICKLIES

A nagging suspicion that somewhere you have overlooked something critical and are going to be blamed for it.

COLD START

Turning the machine on again. Contrasts with **warm start** in which the computer remembers some previous settings or reuses one or more programs already loaded on it (cf **reset** and **ctrl-alt-del**).

COLUMNS

1. A unit of width on a **monitor** or **printer** equal to one character. Most monitors have displays about 80 columns wide.
2. In **RDBMS** jargon, a **field**. Derives from the representation of **databases** as **tables** with columns (the same type of data in each **record**) and rows (the collection of fields which make up one record).

COMMAND

An **instruction** to the computer to do something. Commonly has the form of single words or short phrases which make some kind of sense (at least to the programmer who invented

them). Thus the **MS-DOS** command *del* is short for *delete* and instructs the computer to remove one or more specified files. But, to make sure that you get confused, different **operating systems** and **applications** will use different words to mean the same command or, worse still, the same word to mean different commands. To get you really rattled, programmers will even produce **programs** in which the effect of the same command word will be totally different depending on where you are in the program.

COMMAND LANGUAGE

A kind of programming **language** used to give instructions within **applications software**. Sometimes very specific to that application but some command languages are now common to 'families' of applications (e.g. dBASE). (cf **enquiry language**)

COMMS

Short for 'communications'. Mainly describes **software** which enables one computer to 'talk' to another. Also used to describe **hardware** with similar function. (cf **model**)

COMMUNICATIONS

See **comms**.

COMPATIBLE

Broadly means 'will work with' or 'is very similar to'. Thus a computer identified as 'IBM compatible' is (or is claimed to be) very similar to the IBM family of computers and will use **software** designed to run on those machines.

COMPILER

See **code**.

COMPUTER

A machine which does calculations. Also used as an expletive or the universal excuse for anything going wrong. Also a device for issuing rates bills amounting to £99,999,999.

CONFIGURE

To set up a computer (or **software**) to work in particular ways. For computers, this usually means things like specifying the display **adaptor**, the **port** to which **printers** and other **devices** are attached, **screen** colours etc. For software, the configuration might be the same kinds of things but, more often, it means telling it *which* printer you are using, *where* in your **directories** your **files** are kept etc.

CONTROL-C

A combination of keys which, when pressed simultaneously, will often interrupt the operation of a **program**. (cf **crash**)

COPY

As a verb (*to* copy), means to make an exact duplicate of a **file**, or part of the contents of

a file, or all of the files on a **disk**. Occasionally used to mean **loading** a **program** (itself contained in a file) into memory.

As a noun (a copy), refers to the duplicate produced by copying.

COPY PROTECT

What some **software houses** do to prevent you from stealing their software. When you try to run the **program** from the **copy** it recognises that it is not the supplied version and refuses to work. A real pain if the copy in question is a legitimate one or if something goes wrong with the one you were supplied with. Fortunately most software houses are abandoning copy protection and will now just sue you for millions if you rip off their software. Unfortunately the one company still bucking the trend is the Lotus Corporation, suppliers of the ubiquitous **spreadsheet** called **Lotus 1-2-3**.

CORRUPT(ED)

Scrambled, messed-up, mangled beyond recognition and recovery. Happens to **disks, files** and **databases** and causes pain, anguish and a lot of hard words.

CP/M

An **operating system** used on some **micro-computers.**

CRASH

The unscheduled termination of a **program**. Can, however, be deliberate (as in 'it was time to go home, so I crashed it'). (cf **control-c, fall-over, head crash**)

CTRL-ALT-DEL

A combination of keys which, when pressed simultaneously, will cause the computer to stop what it's doing and start again by reloading the **operating system**. Usually called a 'soft' reset or a **re-boot** (some computers have a special button on them to do the same thing). Generally considered jolly bad form and **end-users** are not supposed to do it.

CURSOR

One who curses (see also **end-user**). Also the square blob or flashing underline on the screen to tell you where you are. If the cursor stops flashing (if it was in the first place), this is often a good sign that the program has hung.

CUSTOMER

Someone who is going to buy some hardware or software but hasn't paid for it yet.

CYA

Pronounced see-why-ay. To guard against being blamed.

DAISY WHEEL

The print-head on particular kinds of printer. Produces high quality print (often called letter quality) but usually in a very limited range of print styles. (cf **nlq, dot matrix**)

DATA

Plural of 'datum' (but going the way of 'media' in being treated as singular) meaning items of information. Sometimes used to distinguish the information a **program** is working on from the program itself. Otherwise used by some (FESC, for instance) to distinguish any old stuff you've been able to collect from information which is actually useful and worth the bother.

DATABASE

A collection of **data** usually in an ordered and accessible form.

DATABASE MANAGEMENT SYSTEM (abbr. DBMS)

A computer **application** for holding your **data** and allowing you to get at them in the way you want. Can be quite simple, like an electronic version of a cardfile (there is, in fact, a DBMS product called just that) or mind-bogglingly complicated. In between are products like **dBase II** and **III** which are reasonably easy to understand. It's only their manuals which are mind-bogglingly complicated.

DBASE II AND III

A **database management system** produced by a company called Ashton-Tate, allegedly the best selling **DBMS** for **micro-computers** in the world. Can be used in a simple way for storing and retrieving data using its own **command language** or can be part of a very sophisticated system with a lot of automatic **processes**. The former requires of the **end-user** a highish degree of familiarity with computers. The latter is frequently designed specifically to enable users to use the system without having to do a computing degree first.

DEBUG

The process of removing all the mistakes in a **program**. (cf **bug, fix, patch, version**)

DEC

Acronym for Digital Equipment Corporation, a big American computer manufacturer specialising in **mainframe** and **mini-computers**.

DECISION SUPPORT TOOL (or SYSTEM)

Fancy term to describe **user-friendly software** designed directly to assist decision making; by filtering, analysing and aggregating data, by presenting them in graphical form and by permitting rapid and easy changes to be made in 'models'. The **DST**, par excellence, is the **spreadsheet**.

DEFAULT

A value or condition used by the computer if not instructed to use something else. Thus the default date can be set to today's date and this will automatically be used by the computer unless you give it another to use.

DESK TOP PUBLISHING (abbr. DTP)

An **application** for producing very high quality print with a lot of control over how it looks. Allows, for instance, many different styles and sizes of print and multiple columns which can be made to 'flow' round irregular shapes (like pictures). In one sense, it is a very clever **word-processor** but DTPs can often also handle graphics (drawings or even digitised photographs). To take advantage of the facilities of DTP you need a very versatile printer, such as a **laser printer**.

DEVICE

Generally, any bit of **hardware**. Simple enough concept until you realise that computers can be made to operate as though there are more bits of hardware than there actually are. The extra devices are sometimes known as *virtual* or *pseudo* or *logical* devices. Confused? Wait until you try to use one!

DIGITAL

Of or pertaining to digits. In physiology, this presumably refers to fingers. In computing, it generally means the kind of computer which works by processing binary digits, i.e. all of them (but cf **analog**).

DIRECTORY

A list of **files** on a storage **device**, usually a **disk**. Useful because it allows files to be grouped together in ways which make them easier to find, particularly when there are a lot of them. Can be thought of as analogous to a category filing system in a library. The directory structure is organised by the **operating system**.

DISABLE

To turn off a feature usually via a **program** rather than physically. Thus, to disable the printer is to arrange in the program that it will not use the printer.

DISK

A round, flat piece of plastic coated with a magnetic medium on which **data** are stored. Will retain data more-or-less indefinitely (more if it is looked after, less if it is used as a coffee-cup mat, is left on top of a radiator or on top of a **monitor**). Looks a bit like an audio record but works more like an audio cassette. Can be **floppy** or **hard**.

DISKETTE

Same as **disk** but only **floppy**, not **hard** ones.

DISK DRIVE

The **device** into which the **disk** is inserted (or permanently mounted) in order that the computer can **read** from or **write** to it.

DONGLE

A **security device**, occasionally used with particular **software** to prevent it being illegally

copied. It plugs into a **port** on the computer and the software won't work without it. Also, an incompetent person (as in 'look what you just did, you dongle!). (cf **copy protect**)

DOS

Acronym for Disk Operating System (see **operating system**).

DOT MATRIX PRINTER

A printer which forms characters on paper by pressing lots of little pins onto the printer ribbon. The pattern of pins produces the shape of the character. Usually lower print quality than **daisy wheel** but more flexible and capable of producing **graphics**.

DOWN

Not working (as in 'my computer has gone down'). Often caused by the computer getting very depressed and refusing to play any more (cf Hitchhiker's Guide to the Galaxy). Fortunately, usually temporary.

DOWNLOAD

To **load data** or **programs** held on one computer (usually a large one) into the **memory** or the **disk** of another (usually small).

DUMB TERMINAL

A **device** — **monitor** and **keyboard** — attached to a big(gish) computer to give access to the computer. 'Dumb' because the **terminal** won't work unless attached to the main computer. An 'intelligent' terminal can function as a computer on its own. Both, however, require fairly intelligent people to operate them.

EDITOR

A very simple **word-processor** mainly used for writing programs. (Alternatively, a highly skilled, trained professional person who transforms the ungrammatical ramblings of others into polished prose.)

EMAIL

Means Electronic **MAIL**. Using computers to provide a communications system. Can operate only locally (via a **local area network**, say) or can use the public telephone system. Examples of the latter are PRESTEL, TELECOM GOLD and TTNS (The Times Network System).

EMIS

The people who make **FEMIS**.

EMULATOR

A means (usually **software**) of getting one kind of computer to pretend to be another kind. Used to overcome incompatibilities. (cf **compatible**)

END-USER

You. The person who actually wants the computer to do something useful. This notion, who exactly is the **software** being produced for, seems to be a tricky one in the computer business. In the past, when computers were wholly the province of technocrats, software was designed by experts for experts to use. The end-user was therefore a computer specialist and lots of things didn't need to be designed in (such as, for instance, any explanation of how the software worked). The advent of the **micro-computer** meant that perfectly normal people were beginning to use computers. They didn't know, or want to know, about bits and bytes and all the technical stuff. They just wanted to be able to use their computers with software which didn't require them to get involved with the technicalities. Hence the notion of user-friendly — software designed for the people who are going to sit in front of the computer and press the keys. Unfortunately, you will still come across too many examples of software written for the staff of your Data Processing Department— which is tough if you don't have one.

ENQUIRY LANGUAGE

A set of commands provided with **applications software** (typically, **database management systems**) to allow the user to get information out of (or 'interrogate') the application. Can be very simple or as powerful and complex as a full-blown programming **language**. (cf **command language**)

ENTER

The big key on the right of the **keyboard** with a bent arrow on it. Used to instruct the computer to execute a command typed in via the keyboard. (cf **return**)

ENVIRONMENT

The OED defines the word to mean *surrounding objects, region or conditions* and this is pretty much what it means in computing too. Thus 'an **IBM** environment' means 'we mainly use **IBM** equipment here' or 'a **COBOL** environment' means 'most of our programs have been written using the **COBOL** programming **language**'. Broadly, the generally prevailing condition or situation.

ESCAPE

A key on the **keyboard** (usually labelled 'ESC') commonly used to stop whatever function you happen to be doing at the time. The precise effect of pressing the key can be determined by the programmer so it will vary depending on the **software** you are using.

EXECUTE

To carry out an **instruction**. Also, to terminate (people).

EXIT

To leave a **program**, or **process** (as in 'exit to **DOS**'), or the room.

EXPERT SYSTEMS

An approach in computing which attempts to imitate human thought processes (or their best guess as to what those processes are). Supposedly includes approaches such as problem solving, rule-based logic and generating rules by induction. (cf **artificial intelligence**)

EXPORT

See **Import/Export**.

FALL-OVER

To stop working. Applies equally to **programs** and programmers (but happens more often with the latter). (cf **crash**)

FEMIS

Stands for Forever Expecting More Inflated Salaries (oh, all right then, it is actually short for Further Education Management Information Systems), a commercially available **Management Information System** specifically designed for further and higher education.

FIELD

A term used mainly in **databases** to refer to an item of information or the place specified on the **screen** in which that item of information is to appear (or be inserted). Several fields constitute a **record;** several records constitute a **file**. (but cf **relational database management systems**)

FIFTH GENERATION

The one not invented yet. (cf **generations**)

FILE

The basic unit of information storage (on, for instance, a **disk**). A collection of coherent (maybe!) information. (cf **field, record, database**)

FILE NAME

A unique identifier of a **file**. **DOS** requires that the name is not longer than eight characters and must start with a letter. Optionally, you can add a further three characters (separated by a full-stop) called the file **extension**. The extension is mainly used to show what kind of file it is. Files which end in .COM or .EXE are executable files, i.e. they contain **programs**. Files which end .TXT or .DOC are usually text files, i.e. if you print them out, recognisable text will be produced. Many **applications** automatically add their own extensions to files created by the application as a way of recognising them.

FILE SERVER

A **micro-computer**, with a **hard disk** in or attached to it, linked to several other micro-computers in a **local area network**. The file server acts as a central store of common

programs and **data** and controls the access (or sharing) of the common facilities among the other computers (often called, in this context, **stations**).

FIRMWARE

Half way between the **hardware** and **software**. A term which proves beyond reasonable doubt that computer people have little better to do with their time than dream up obtuse, unnecessary jargon. Supposed to refer to software permanently (or semi-permanently) **installed** in a computer (usually in **ROM**).

FIX

A piece of **code** correcting one or more **bugs**. (cf **patch, version**)

FLOPPY DISK

A small removable **device**, which looks a little like a record but works more like a cassette on which a computer stores information magnetically. Available in a range of standard sizes and not necessarily floppy. (cf **disk, hard disk, winchester**)

FONT

Large bird bath found inside churches, for washing babies in. Also the particular shape and style associated with text. This text, for instance, is 'Times Roman'.

FOOTPRINT

Like a fingerprint but bigger. Also the base area (the amount of space on your desk it will take up) of a piece of equipment.

FORMAT

Describes the way in which a computer organises its storage on **disks**; different systems adopt different formats and just because a disk will physically fit in a machine it cannot be assumed that it can be read. Also used as a verb — to format (sometimes, 'initialise') means to prepare a disk to receive information. Also describes the way **applications** and **operating systems** organise the **data** within a **file**. (cf **import/export**)

FOURTH GENERATION (LANGUAGE OR ENVIRONMENT)

See generations.

FRED

When programmers are testing anything, they always (well, maybe not always) call it fred . Thus test and demonstration **files** are commonly called fred. Ashton-Tate (see **dBASE**) have invented a **command language** for a product and they have called it (yes, you guessed it) FRED ! Also, the name of programmers' cats. (cf **kit**)

FRETWELL-DOWNING

The other people who make commercially available **Management Information Systems** for FE. The product is called College Management Systems (CMS).

FRONT-END

The opposite end from the back-end. A term sometimes used to describe **menus** and other **user-friendly interfaces** put on the front-end of less friendly **programs**.

FUNCTION KEYS

Most computers use an expanded **keyboard** which, in addition to the letters of the alphabet and numerals, includes special keys which serve special purposes. A group of these are the function keys, usually shown as F1 to F10 or F20 , which can most often be found along the top of the keyboard or in a group on the left.

GENERATIONS

A way of distinguishing the various phases in computer **hardware** and **software** developments since the 1940s. First generation software is **machine code**, the second is **assembly language** and the third is **high level languages**. The **fourth generation** is (allegedly) the programming (or **command** or **enquiry**) language used with **relational database management systems**. An example of this last is **SQL**, used by **ORACLE** and **EMIS**'s student record system.

GIGABYTE

A hell of a lot of bytes (or, if you insist on being technical, 1,000,000,000 bytes — approximately).

GIGO

Garbage In — Garbage Out.

GLITCH

Technically a short but fatal interruption in the power supply; but also any problem you are trying to pretend is not very important.

GRAPHICS

Any display of information on **screen** or **printer** that is not simply text or number. Often requires a special **adaptor** and sometimes a special **monitor** or **printer**.

HACKER

Cabinet, turned Prime, Minister or computer enthusiast with a predilection for trying to overcome computer **security** systems. Verb — to hack.

HALT

To stop, cease working. (cf **hung, fall-over, lock, crash**)

HANDSHAKING

The consequences of drinking too much coffee (cf **coffee**). Also the means by which computers control the operation of other **devices**, especially **printers**.

HANG

See **hung**.

HARD COPY

Computer **output** from a **printer** or similar **device**. Also, very difficult books (i.e. for programmers, anything with words in).

HARD DISK

A storage **device** which is permanently built into a machine, unlike a **floppy disk** which can be removed and carried about. Hard disks, sometimes called **winchesters**, typically have much greater storage capacity than floppies — upwards of 20 Mb. (cf **disk, megabyte**)

HARDWARE

The physical apparatus which goes to make up a computer — the boxes and electronic bits as opposed to the **programs** or **software**. (cf **kit, software, program**)

HEAD

The specific bit of equipment through which the computer conveys information to and from the **disk**.

HEAD CRASH

A catastrophic event in which the **head** gouges a permanent and expensive trough in a **disk** (usually while the computer is being thrown around). The really upsetting thing is not being able to get at any of the months- or years-worth of work stored on the disk. It is roughly equivalent to welding a full filing cabinet permanently shut. It is usually the point after which people start taking **back-ups**.

HIGH LEVEL LANGUAGE

A programming **language** which bears some resemblance to ordinary English. Still very cryptic, however, and you need to be an expert to understand it. Fortunately computers do (well, sometimes they do). Examples are **BASIC**, C, **COBOL**, and FORTRAN. (cf **machine code**)

HUNG

The exquisite experience of typing 25 pages of your PhD using your **word-processor** and discovering that the machine refuses to do anything at all. It just sits there staring back at you. No amount of frantic key pressing gets any action out of the demonic device (although it might bleep at you a lot). The only thing you can do is TURN IT OFF, thus losing your insights forever! (cf **crash, lock, save**)

IBM

Said to stand for 'I've been moved' (a reference to their personnel policy). Otherwise, International Business Machines, the world's biggest computer company.

ICON

A visual representation of a function or **program** (i.e. a friendly little picture of it). Used in so-called **user-friendly interfaces**. (cf **windows**)

IMPORT/EXPORT

Moving **data** out of one computer system and into another (and vice versa). The exact form in which data are held will vary according to the **applications software** and the **operating system**. The facility to get data collected by one application into another is a vital part of **compatibility**. Most software provides some means of writing data to a **file** in a form which can be **read** by other **software**. There are a number of standard **formats** for export files. The common ones are **ASCII**, DIF (Data Interchange Format) and Comma Delimited Format.

INK-JET PRINTER

See **printer**.

INPUT/OUTPUT

That bit of the **operating system** dedicated to passing instructions to the **CPU** from an input **device** (like a **keyboard** or a **disk**) and passing the results to an output device (like a **monitor** or a **disk**). Also refers to a particular chunk of the electronics inside the CPU box.

INSTALL

Means anything from getting the **hardware** out of its box and putting the various bits together to **copying** a **file** onto a **disk**. Big(gish) computer systems (say a **mini-computer**) are complicated beasts to set up. Not only does the hardware require expertise to put together but the **operating system** and the **applications software** have to be persuaded to work. Some software is installed just by putting the **floppy disk** in the appropriate **drive**, or by copying it onto a **hard disk**. Often, however, more complicated things have to be done to **configure** the software to your computer. (cf **warm start**)

INSTRUCTION

Generally, a synonym for **command**. Can also mean one line (a single element) of **code** in a **program**.

INTEGRATED CIRCUIT

Computers used to consist of many components like transistors, capacitors and resistors. An integrated circuit combines hundreds or thousands of these on one small piece of material (which used to be silicon — hence **silicon chip**) smaller than a postage stamp.

INTEGRATED SOFTWARE

A linked collection of different kinds of **applications software**, usually all from the same company. Typically consists of a **DBMS**, a **spreadsheet** and a **word-processor**.

INTERACTIVE

A mode of communication with a computer in which the **instructions** issued by the operator are acted on immediately. Other terms meaning the same thing are **'on-line'**, 'live' and 'real-time'. Contrasts with **batch**.

INTERFACE

Anything which connects anything with anything else. Mainly a posh term for a bit of cable (presumably to justify its absurdly high price) but also used to mean the way in which **software** presents itself to the **end-user**. (cf **user-friendly interface, wimp, windows**)

INTERPRETER

See **high level languages** and **code**.

I/O

See **Input/output**.

K(b)

Kilobyte. A thousand (more or less) **bytes**. A very large mouthful. Also Kilobucks — the measure of programmers' salaries.

KEYBOARD

The thing with the keys on it. Contains standard typewriter keys (the **'QWERTY'** keyboard) plus a lot of others with particular purposes. These others include **function keys** and the numeric **keypad**.

KEYPAD

Usually refers to additional keys on many computer keyboards which often have two functions. The first is an alternative way of entering the numbers 0 to 9. The second is to move the **cursor** around. These keys are, therefore, sometimes known as the cursor keys.

KILOBYTE

See **K(b)**

KIT

An umbrella colloquialism for all kinds of computer **hardware**. Also, what programmers' cats are called (if they are not called **Fred**).

KLUDGE

A bodge, a lash-up. A particularly inelegant solution to a problem. (cf **bug, fix, patch**)

LANGUAGE

The particular words and phrases (with a greater or lesser resemblance to English — or another language) which can be translated by the computer into instructions (in **machine code**) which it can then **execute**. (cf **binary, code, high level language, machine code**)

80

LAP-TOP

Term used to describe small, light, portable computers. The name is intended to indicate that you can operate them while resting them on your lap without breaking your ankles. Usually found in the possession of yuppies and accompanied by copies of 'Investors Chronicle' and a portable telephone. A more expensive and heavier version of a **Filofax**.

LASER PRINTER

A **printer** which works (and looks) much like a photocopier. Faster than a normal printer and can handle a wide range of typestyles and other images (such as **graphics**). Produces very high quality print and is therefore often used with **desk top publishers**. Were very expensive but are getting cheaper. Reasonably safe to use providing you are completely protected by aluminium foil (shiny side out).

LCD

Stands for Liquid Crystal Display. Now ubiquitous on the front of digital watches but also used to make the flat, space-saving **monitors** for **lap-top** computers.

LED

Short for Light Emitting Diode. Previously ubiquitous on the front of digital watches (see **LCD**), but now more commonly used for the little lights on the front of computers to kid you into thinking that they are working.

LIST PROCESSING

See **Mail Merge**.

LOAD

To put a **program** or **data** into the **memory** of the computer prior to doing something useful with them. Thus, the first thing a computer does is load the **operating system** program into its memory. (cf **process**)

LOCAL AREA NETWORK (abbr. LAN)

A group of **micro-computers** (called, in this context, '**stations**') linked together by a cable, able to communicate with one another and (usually) sharing certain facilities like **printers** and one or more **file servers**. 'Local' because the stations are usually fairly close together (e.g. in the same building). (cf **network, wide area network**)

LOCK

Has three meanings;

1. Technical term for protecting a part of a **database** from being altered by two users at the same time — necessary to prevent the database from being **corrupted**.
2. Vernacular alternative for **hung**.
3. Key for typing continuous capitals (usually labelled 'shift' or 'caps' lock).

LOGIN (or LOGON)

Another of those vague words. Refers to the process of getting past the first level of **security** usually by entering a **username** (otherwise known as the 'logon id') and, commonly, a **password**. By extension, used to mean getting into the computer system, however you do it.

LOTUS 1-2-3

A particular and very common brand of **spreadsheet**.

MACHINE CODE

The '**language**' a computer can actually do something with and into which all other programming languages have to be translated. The translation is carried out by a **program** (called a **compiler** or **interpreter**). (cf **assembly language, binary, code, high level language**)

MACRO

A sequence of instructions, inside an **application**, which can be given a name, stored away and then **executed** by using the name or a particular key combination (this is sometimes called a 'hot key'). At its simplest, it's a way of avoiding the need to enter an often-used series of instructions (these are sometimes known as 'autos') but in more complex forms are themselves a type of programming. Most frequently found in **spreadsheets** and **word-processors**.

MAIL MERGE

Otherwise known as **list processing**, this is the facility to combine selectively two or more pieces of text. Its most common usage is in **word-processing** when a standard letter, say, can be customised with individual names and addresses. Thus the individual letter you — and four million others — got from Readers' Digest (or the Consumers Association or the AA or. . .) was produced using mailmerge techniques. The customising elements (e.g. the name and address) can be extracted automatically from a **database** thus further depersonalising the apparently personal letter you got from the Managing Director.

MAINFRAME

Once upon a time a mainframe computer was a real one and all the rest were toys. It is big, powerful, can support many **terminals** at once and can run large numbers of **programs** apparently simultaneously. However, it also requires a special environment (e.g. air conditioned), an army of white-coated acolytes to attend to its every whim and costs the earth. Fifteen years ago, if you wanted a computer, that's all there was. These days, **mini-** and **micro-computers** have the capabilities of those mainframes (but, of course modern mainframes are bigger, more powerful etc).

MANAGEMENT INFORMATION SYSTEM (abbr. MIS)

A system (usually, but not necessarily computer based) processing information relating specifically to the management of the enterprise. Typically a **database management**

R2

system concerned, in the commercial world, with analysed financial data. Term used, by extension, to refer to the total computer-based information system.

Mb

A **megabyte**.

MEGABYTE

More bytes than a **kilobyte** but less than a **gigabyte**. Mega means million. Enough clues?

MEMORY

Most often means that bit of the computer's insides into which the **programs** and **data** are loaded while they are being processed. Sometimes called **RAM**, it will only hold its contents while the electricity is on — it is therefore 'volatile' and not a suitable parking place for anything you want to keep. The power of **mini-** and **micro-computers** is often given in terms of their memory sizes. The BBC B micro-computer was 32 Kb; a typical business micro-computer today will have 512 Kb or more in its memory and memories of 4 Mb and greater are now quite common.

MENU

A list of choices from which the **end-user** selects one option. That selection often produces another menu (which may be called a sub-menu) offering another range of choices. Intended to increase the **user-friendliness** of **programs** by reducing or removing the need to know and understand how the program works. (cf **user-friendly interface, wimp, windows**)

MICRO-COMPUTER

Probably originally meant a computer which was based around a single **micro-processor** (cf **integrated circuit**). Now used in a forlorn attempt to distinguish one kind of computer from other kinds, i.e. mini-computers and mainframes. Loosely speaking, a micro-computer can only be used by one person at a time (but see **super-micros**) and is slower, smaller, cheaper and less powerful than other kinds of computer.

MICRO-PROCESSOR

See **integrated circuit.**

MINI-COMPUTER

Half way between a **micro-computer** and a **mainframe** — but the boundaries are increasingly blurred. Can be used by several people at the same time (but fewer than a mainframe). Intermediate in terms of its power, speed, size and price but some mini-computers are cheaper and smaller than some micro-computers. A key distinction is that all mini-computers are **multi-user** and **multi-tasking** whereas micro-computers aren't (except some of them!). (cf **super-micro**)

MODEM

Stands for 'modulator/demodulator'. A **comms device** for converting the **digital** output

83

signal from a computer into an **analog** voice-frequency signal which can be transmitted along a telephone wire, and then unconverting it at the other end. To link two or more **remote** computers together (in, say, a **wide area network**), each computer needs a modem. If you find other aspects of computing confusing, you should try scrambling your brain understanding the arcane science of comms and modems!

MONITOR

The **output device** like a TV screen also known as a Visual Display Unit (**VDU**) and a **screen**. There is a bewildering variety of monitors but some of the principal distinctions are:

- Colour or not — Colour or monochrome (not necessarily black and white).
- Resolution (how much detail the monitor can display) — Resolutions tend to be classed as high, medium or low but this is sometimes referred to as certain 'standards': TTL (means low), CGA (colour graphics adaptor), EGA (enhanced graphics adaptor) and now VGA (probably stands for very something-or-other). The '**adaptor**' bit refers to the fact that, to use high-resolution monitors, you need a special **device** (or **card**) inside the computer, called an adaptor.
- Size — Monitors are generally between 12in and 14in in size (i.e. diagonally across the screen) but you can get bigger ones, some of them much bigger. They are also a much bigger price.

The above points are important (except, maybe, size). Some **software** will only work properly with a particular range of monitors and adaptors; almost all software will ask you when you install it what kind of adaptor you've got in your machine. To complicate life further, quite a lot of computers have, or can pretend to have, several different **adaptors** in them and you may have to **configure** them for one adaptor or another to work with particular software.

MOUSE

A member of the **WIMP** family of user **interfaces**. A small **device** attached to a computer which when moved over a flat surface causes a **pointer** to move on the **screen**. The pointer can select options from pull-down **menus** and be used variously to issue **commands** to the computer. A similar device, doing much the same thing is called a **tracker-ball**. Called mouse because it is supposed to look like one. Moves in units called mickeys (approx 2.74 mm). No female version available. Rarely furry. (cf **window, windows, user-friendly interface**)

MS-DOS

Stands for MicroSoft Disk Operating System. A very common **operating system** for **micro-computers** made by the Microsoft Corporation.

MULTI-TASKING

Able to carry out several **processes** (or **tasks**) at once. A characteristic of **mini-computers** and, now, **super-micros**. Actually, all computers can only do one thing at once (but cf

parallel processing) but some of them can switch very rapidly between different tasks so that they look as though they are all happening simultaneously.

MULTI-USER

Able to support several users at once. From the computer's point of view (which is, of course, a very intelligent one), multi-user is not much different from **multi-tasking**. Each user represents a task (or **process**) to the computer and the same considerations apply.

NETWORK

One of those irritating terms in computing jargon which has lots of different and constantly changing meanings. At its most general, it seems to mean any configuration of computers which are or can be connected together. When talking about **micro-computers**, it tends to be used to refer to **local** or **wide area networks**. But it can mean any grouping of computers, including **mainframe** and **mini-computers**, linked together. Is often technically quite difficult to achieve and totally impossible for the average **end-user** to understand.

NIBBLE (NYBLE)

Four **bits**. A small **byte**. Honest! Would we lie to you?

NLQ

Stands for 'Near Letter Quality' and means something like 'almost as good as a decent typewriter'. Refers to the density and precision of the characters produced by a **printer**, usually of the **dot matrix** variety.

NULL

The total absence of anything. Zilch. Nothing. This is not the same as zero, which is something (it is, in fact, zero).

ON-LINE

Connected and switched on. The little light on the front of most **printers** labelled 'on-line' means nothing more than that. Also means **interactive**.

ON-LINE HELP

The facility to get instructions on the **screen** on how to use an **application** while you are actually using it. Saves you having to look everything up in the instruction manual all the time. The help may well appear in a **window** and is invoked by pressing a particular key or key combination (the F1 key is a favourite). 'Context sensitive' help can tell what you are doing in the application and give you instructions relevant to that particular function.

OPERATING SYSTEM

When a computer is switched on, it can do nothing more than check that it's working properly (a self- or diagnostic check) and then look for an operating system usually from a **disk**. The operating system is a set of **programs** which acts as a kind of 'middle-person' between the **hardware** and the **applications software**. In **micro-computers**, the operating

system is quite simple and mainly handles **I/O** operations. **Mini-computer** and **main-frame** operating systems are much more complex as they have to handle **multi-user** and **multi-tasking** operations as well. Operating systems serve to standardise the **interface** between the **hardware** and **software** thus making it possible (at least, in theory) for software designers to produce programs which will work on any machine using that operating system. Operating systems have their own set of **commands** which are mainly concerned with handling **files** and the **directory** structure. (cf **badly behaved, port, well behaved**)

ORACLE

A particular brand of **relational database management system** (**RDBMS**) produced by the Oracle Corporation. Nothing to do with the teletext system run by Independent Television.

OUTPUT

Anything which actually manages to emerge from your computer — usually in the form of images on a **monitor** or print on a **printer**. Can, however, be in the form of a **file** on a **disk**.

PARALLEL

See **port — 1**.

PARALLEL PROCESSING

Computers can only do one thing at a time. Each **instruction** has to be **executed** in sequence. Even those computers which appear to be doing several things simultaneously, in fact aren't. What they actually do is a little bit of each **program** (or **process**) but switch between them so fast that it looks like they are all happening at once. Parallel processors, however, do literally do several things simultaneously. They act as though they are several computers each carrying out at bit of the program (in fact, they are several computers, all linked together). However, many programs logically require that one instruction is executed before another. Others are just written that way because that's how you had to write programs. The first sort can't really be done on parallel processors, the second sort can but have to be re-written in a different programming **language**. Examples of the second sort are weather forecasting **applications** and complex **graphics**. (cf **processing, code, transputer**)

PASSWORD

A 'string' of **alphanumeric** characters which has to be **entered** at the **keyboard** (usually in conjunction with a **username**) to allow the user access to a computer or to a particular **application**. To be properly **secure**, should not be guessable by others. However, it also has to be easily remembered by the user; two mutually incompatible requirements.

PATCH

A 'repair' for a **bug**.

PATHNAME

The 'full' name of a file includes both its actual name (e.g. FRED.TXT) and its location on a **drive** and in the **directory** structure. Thus the full pathname (for MS-DOS) might be C:\DOCUMENT\FRED.TXT, where C: is the drive, DOCUMENT is the directory and FRED.TXT is the filename. The backslash (\) indicates, to **MS-DOS**, links between directories. The computer often has to be told the exact location of a file to find it.

PC

Acronym for 'personal computer' coined in ancient times (i.e. pre-1980) when they were things you bought for personal use, probably at home and probably to play computer games on. Nowadays, it is an alternative term for **micro-computer**.

PC-DOS

Short for 'Personal Computer — Disk Operating System', the **operating system** for the IBM-PC micro-computer. Virtually identical with **MS-DOS**.

PERIPHERAL

An obsolescent term meaning any **device** which was connected to a computer, e.g. **printers, disk drives**.

POINTER

The 'P' in **WIMP**. A little arrow or other shape on the **screen** which you move around using a **mouse** and which can be used as an alternative (and allegedly friendlier) way of issuing **instructions** to the computer.

PORT — 1

As a noun, usually refers to one of the sockets on (or in) a computer to which something else can be connected. Come in two main types — **serial** and **parallel** (other terms for the same kind of thing are RS232 and Centronics). The terms refer to the way in which the signals are transmitted and received. Even experts don't understand the inner mysteries of these connections and trying to can seriously damage your health. (cf **comms, modems**)

PORT — 2

As a verb, refers to the process of transferring **software** from one computer to another (usually of a different type). Probably first referred specifically to the process of rewriting software so that it would work on another type of machine (less difficult if the software is **well behaved**) but now used loosely to mean moving anything.

PORT — 3

Places ships go in storms.

PORTABILITY

Able to be moved. Can refer to the physical bulk of **hardware** but more commonly used about **software** and the ease with which it can be transferred from one type of computer

to another. Generally, the more of it the better. (cf **port — 2, well behaved**)

PRINT

The straightforward meaning is to send the **output** of a **program** to a **printer**. However, you will occasionally come across the notion of 'printing to the screen' — this means nothing more than 'stick it on the screen please'. Also the product of printing, i.e. the words on a piece of paper.

PRINTER

A **device** for producing **hard copy**. Come in an enormous range of types, sizes and makes. Most common types are **dot matrix, daisy wheel, laser, ink-jet** and **thermal**. Some of the most common makes are Epson, IBM, HP (Hewlett-Packard), Oki and Brother but there are dozens of others. Prices range from £100-200 to thousands of pounds. Printers are tricky beasts. They go wrong more often than anything else and they are all different. Some **software** will only work with a limited range of printers and you can get some very funny results if you have not set up (**configured**) the software or the printer properly.

PROCEDURAL

Computers are, generally speaking, not very bright. To get them to do anything, you have to tell them each step of what it is you want them to do; you can't leave any step out and you can't use them in the wrong order. If you do, they go wrong. **Programming languages** (up to and including **third generation** languages) tend to have the same step-by-step characteristics. They are known as procedural languages — because you have to instruct the computer at each stage of the procedure to be followed. This is not much like the way normal human beings think, which is one of the reasons computer programming seems so difficult. **Fourth generation languages** (and some others) are different. With these, you tell the computer what you want to accomplish; it then works out how to do it. Such languages are sometimes known as 'object orientated' (i.e. you specify what the object of the exercise is). It is said that these are easier for lay-people to understand and use, but hard for programmers trained in procedural languages. Actually, of course, they are all incomprehensible!

PROCESS

As a verb (to process), this generally refers to the actions of a computer in **executing** the series of **instructions** given by you or, more commonly, by the **program** you are running.

As a noun (a process), it refers to one of several programs which the computer is trying to run, perhaps more or less simultaneously. It has less significance in **micro-computer** terms as they can only run one program at a time (but cf **super-micros**), but **mini-computers** can handle several operations 'simultaneously', each operation is known as a process. (cf mini-computer, micro-computer, **parallel processing**)

PROGRAM

A series of **instructions** to a computer, usually in a programming **language** and usually brought together in a **file** and given a name. The program can be **executed** by just typing the name. Actually, what you are doing here is instructing the **operating system** (itself a **program**) to **execute** another **program**. And no, it shouldn't be spelt 'programme'; that's

something on the telly (or, for the very couth, something which tells you what's going on at the Opera). (cf **code, machine code, high level language, badly behaved, well behaved**)

PROMPT

The stuff you see on the **screen** which tells you that the computer or the **application** is working and waiting for a **command**. The 'DOS prompt' looks something like this; A:> This tells you that you are currently using the **disk drive** identified by the letter 'A', the machine is working and waiting for an instruction but you haven't any other application or program running. If you did have, it would probably give you a different prompt.

QUERY LANGUAGE

See **enquiry language**.

QUIT

To terminate a **program** or **process** in an orderly manner. Contrasts with **crash**. (cf **exit**)

QWERTY

The name given to the standard **keyboard** on English and American typewriters and also used on computer keyboards. Refers to the sequence of letters with which the alphabetic part of the keyboard starts (in the top left). Apparently designed to be illogical and hard to use — to slow down typists using old mechanical typewriters who could press the keys faster than the machine could respond.

RAM

What Sylvester Stallone is called by his close friends. Also, short for '**R**andom **A**ccess **M**emory'; for which, read **memory**.

RE-BOOT

See **boot**.

READ

To look at the contents of a **file** (stored on a **disk**) or to take the contents of the file and put them in the computer's **memory**. In fact, you can only look at a file by putting them in memory first. Also known as **load** and contrasts with **write**.

READ-ONLY

A condition (or attribute) of a **disk** or a **file** on a disk. Means that you can look at, and/or **copy**, the files but can't change or delete them. Also known as **write protect**. Used for protecting important files or disks from accidental changes or deletion. Disks are set read-only by sticking a piece of tape over a notch in the edge of the outer case or by moving a special tab.

RECORD

A collection of **data** about the same subject. Several records make a **file**. (cf **DBMS, field**)

RELATIONAL DATABASE MANAGEMENT SYSTEM (Abbr. RDBMS)

A particular kind of **DBMS** in which any item of data contained within it can be linked to any other data item. This is known, in the jargon, as 'data independence'. You might think that this would be a characteristic of all databases, but it isn't. One consequence of data independence is that RDBMS's can be modified, extended, etc. much more easily than other kinds of DBMS. In theory this means that you don't have to have decided, before you start, exactly how your database should be constructed as you can come along later and change it. Unfortunately, the system of which the database is a part consists of a lot more than the database itself. The other bits may well be very difficult and expensive to change.

RDBMS's have their own jargon. A **file** is called a **table**. A **record** is called a **row** and a **field** is called a **column**. Table, row and column describe how the RDBMS concept looks on paper but, as you would expect, this is nothing to do with reasons for inventing these terms in the first place. (cf **fourth generation environment**)

REMOTE

At a distance from. Thus a remote **terminal** is one connected via some intervening gadgetry, such as a **modem**, to another computer. Gets confusing because the physical distance is not particularly relevant. A remote terminal could be located right next to the computer to which it is linked — the significant point in this case is that the terminal is not plugged directly into the 'host' computer.

RE-SET

See **boot**.

RESOLUTION

See **monitor**.

RETURN

The key on the right of the standard **keyboard** with the bent arrow and the word **Enter** or **Return** on it.

ROM

Stands for 'Read-only Memory' and refers to a section of the **memory** in the computer's **CPU** in which one or more **programs** are permanently stored. 'Read-only' because the **end-user** can't normally put anything in there or delete its contents. Usually contains the program which the computer uses to check for **hardware** faults and which tells it where to look for the **operating system**. Sometimes (but rarely) contains the operating system itself. The memory and, more particularly, the software in it are sometimes called **firmware**.

ROW

See **relational database management system** and/or **spreadsheets**.

SAVE

To keep something more or less permanently by **writing** it to a **file**. Something it is very important to do as computers have a nasty habit of packing up three quarters of the way through a long and difficult job. (cf **hung, lock**)

SCREEN

See **monitor**.

SCROLL

A rolled up papyrus containing great thoughts from the past, usually in Hebrew. Also, the movement of text (or other image) on a **monitor** so that it rolls smoothly up, down, across or off the screen, rather than a screenful at a time.

SECURITY

Comes in three main types:

- Physical — keeping the computers, or **disks** locked away so that malevolent or careless persons can't get at them.
- Access — keeping people not authorised to use the system, or part of it, out. Can be physical but tends to use **passwords** and **usernames**. Modern **software** can permit very sophisticated distinctions so that you can, for instance, deny particular people access to a very small bit of a **database**, or you can allow them only to **read data**, or only amend them or vice versa or any combination of the above. Organising access security can be a full time job as it is now so complicated.
- System — protecting the system from failure caused by, for instance, someone pulling out the mains plug. A secure database is one which can recover from such a failure without losing much of the data and without **corrupting** the data.

SERIAL

See **port — 1**.

SILICON DISK

Normally the computer's **memory** is occupied by **programs** and by the **data** being used by the programs. The program, together with the **operating system**, manages the memory and the **end-user** has no direct control over it. You can, however, reserve part of the memory for storage of **files** etc as though it were a conventional **disk**. The advantage of doing so is that it makes the computer work much faster (the slowest part of the computer is usually the disk drive because it is mechanical). The disadvantage is that anything stored in a silicon disk will be deleted when the machine is turned off. Also known as a **RAM or disk drive**.

SOFTWARE

The **instructions** the computer **executes** rather than the computer itself. The collective term for any kind of **program**. Contrasts with **hardware**. 'Soft' in computer jargon generally means 'impermanent' or 'changeable' and there is a trend toward controlling as much as possible of the operation of a computer via software as this makes them more adaptable.

SOFTWARE HOUSE

Any company or other organisation designing **software** and trying to sell it. There are all kinds of software houses and the industry has a different term for each kind. It's not worth bothering with these categories (unless, of course, you are looking for a job in the business).

SPREADSHEET

One of the most useful bits of **applications software** and one of that group known as **decision support tools**. A means of holding, analysing and displaying all kinds of **data**. Holds data in **cells** arranged in **rows** and **columns**. The really useful thing about spreadsheets is that you can establish relationships between different bits of data and automatically calculate the results. You can then change the data and recalculate the new result without having to do the sums again. Doesn't have to be figures; you can do all kinds of whiz-bang things with text as well. Generally comes with graphics so you can display your results as charts and graphs.

SQL

Stands for Structured Query Language, a **Fourth Generation Language**. Originally a research project for **IBM**, it is probably the most widely used of the fourth generation languages. Non-**procedural** (well, almost), very powerful and rather easier for ordinary mortals to use and understand than earlier programming languages. Frequently used as the **query language** with **Relational Database Management Systems** such as **ORACLE**.

STAND-ALONE

Any computer which is operating independently of any other, is called a stand-alone. Some stand-alone computers can be connected to other computers, in which case they become **network stations** or **remote terminals**. Contrasts with **dumb terminal**.

STORE

Same as **save**.

SUPER-MICRO

Half way between a **micro-computer** and a **mini-computer**. Generally speaking, super-micros use micro-computer software but can be multi-user (typically about four users). Bigger, faster, more powerful and more expensive than a micro-computer but smaller, slower, less powerful and cheaper than a mini-computer. Got it?

92

TABLES

The way in which **database files** are represented in **relational database management systems**. (cf **column, record, row**)

TERMINAL

The condition of a computer after you drop it a few times (as in 'this machine **has gone into** terminal failure'). Also, a **keyboard** and a **screen** attached to a computer. Can be **dumb** or 'intelligent'. When an ordinary **micro-computer** is attached (possibly via a **network**) to another computer, the micro-computer is known as a terminal.

TRANSPUTER

The brand name of a particular **parallel processing device**.

TURNKEY

A sales ploy in which it is claimed that the computer system just has to be unpacked and turned on to work properly and completely — as in 'you just turn the key and it goes'. This is almost always an outright lie. Also refers to **bundles** of complete **hardware** and **software** packages.

UNDOCUMENTED FEATURE

A **bug** which the **software house** is trying to represent as a jolly good thing.

UNIX

An **operating system** for **multi-user computers**. Rapidly becoming the industry standard for **mini-computers**.

UP

Working. The opposite of **down**.

UPDATE — 1

To amend a **file** or (more often) a **database** to include the latest **data**. Also used to mean any kind of change.

UPDATE — 2

A new improved version (or 'release') of some **software** or **hardware**.

UPGRADE

A bigger, newer, faster, better one. Alternatively, one which now works more or less correctly (unlike the previous one). (cf **bug, fix, patch, update — 2, version**)

USER-FRIENDLY

Slightly understandable to the average human being. Term of abuse among programmers.

User-friendly **interfaces** include **windows**, **menus** and **mice**. (cf **WIMP**)

USERNAME

Part of an access **security** system. A string of **alphanumeric** characters which the user has to **enter** at the **keyboard** before being allowed further access to the computer system. Commonly used in conjunction with a **password**.

UTILITY

Programs for use in the care and maintenance of computing systems — sometimes called 'house-keeping' rather than particular applications. An example is a program for **locating** a named **file** wherever it is on one or more **disks**.

VALUE

The contents of, for example, a **database field** or a **spreadsheet cell**. Any number or piece of text supplied by the user rather than the programmer. Sometimes called a variable.

VAX

The name of a series of computers made by DEC. Originally **mainframes** and **mini-computers** but now also **micro-computers**.

VDU

Short for Visual Display Unit. See **monitor**.

VERSION

Usually a way of keeping track of the various editions of **software** (e.g. version 1.1). Important because earlier versions are not always **compatible** with later ones of the same software and, more to the point, not compatible with other software. Most common problem occurs when a later version of **applications** software won't work with earlier versions of an **operating system**. Sometimes, a way of getting the customer to pay for a big **patch**. (cf **bug**)

VLSI

Short for Very Large Scale Integration, a particularly powerful **integrated circuit**.

VMS

The name of the **operating system** used on DEC mini- and **mainframe computers.**

WARM START

Loosely means 'starting part-way through'. Mainly applies to **programs** which have to be set-up (or **installed**) in particular ways when first used. They are warm started if you don't have to set them up all over again. By extension, used to refer to any **process** which doesn't have to be begun again from the beginning (for instance, **re-booting** without actually turning the computer off).

WELL BEHAVED

The opposite of **badly behaved**. **Software** which directs all its dealings with the hardware through the **operating system**. Supposed to standardise software so that it will work properly on any computer using the same operating system. Nice theory but often falls down in practice. The first problem is that designers produce badly behaved software largely because it tends to work faster or better than well behaved **programs** — the price is that it will only work properly on the particular machines for which it was designed. A second problem is that there is no real standard method for handling some things, such as the **monitor** display. In consequence, anything that does clever things with the display (e.g. most games), may not work properly on other machines even though it is doing everything else right.

WIDE AREA NETWORK (abbr. WAN)

Any group of computers linked together over any distance and able to communicate with each other. 'Wide' to distinguish it from '**local**'. The distances are usually quite large (e.g. miles apart or even in different countries) and a WAN usually includes **communications devices** such as **modems**. (cf **network, local area network**)

WIDGET

A term for any piece of electronic gadgetry — usually the stuff inside a computer (also known as gizmo, thingumajig, or dodad).

WILD CARDS

A special character recognised by **operating systems** to mean 'and anything like it'. The common ones are * and ?. Thus *.txt stands for any file ending in .txt — the * represents any number of other characters. Similarly, ? stands for a single character, so fred.ba? will find files called fred.bat, fred.bak, fred.bas and so on. Very useful for locating, **copying** etc. groups of similar **files**.

WIMP

Stands for **Windows, Icons, Mice** and **Pointers** (or Pull-down or Pop-up **menus**). The various elements of so-called **user-friendly interfaces**.

WINCHESTER DISK

Another name for a **hard disk**.

WINDOW

Usually, one of several areas of the screen containing the output of different parts of the same **application** or of several different applications. Thus **spreadsheets** can be windowed so that the beginning and the end of a very long spreadsheet can be examined at the same time or two or more different spreadsheets can be seen on the screen simultaneously. Can give the impression of **multi-tasking**, i.e. several **programs** running at the same time. This isn't actually what is happening as only one of the programs is active, the others are lurking in the **background**.

WINDOWS

A product of the MicroSoft Corporation (designers of the **MS-DOS operating system**) which enhances their operating system and makes it easier to use. Uses **WIMP** and allows the user to have several different **applications** on screen at the same time. Includes some so-called Desktop Applications such as a calendar/diary, a clock, a notepad and a calculator. The theory is that the busy executive (!) will use these applications instead of the traditional Filofax, calculator etc. Filofaxes are, however, easier to carry around in your briefcase than a micro-computer. Another example of the same thing is **GEM** and the operating system on the **Apple** Mac.

WORD-PROCESSING

An **application** designed specifically to process text. Sometimes a whole computer is given over entirely to handling text (often called a dedicated word-processor). Hundreds of different varieties available, each with a different range of facilities — from the very basic (which might be called **editors**) to the sophisticated with facilities such as spell checking, a thesaurus, outlining, automatic indexing, table creation and automatic paragraph renumbering. Most word-processors have word-wrap, header and footer creation, and automatic page numbering. There ain't no such thing as the 'best' word processor; the 'best' one for you is the one that does what you want. Common examples are WordStar, NewWord, DisplayWrite, LocoScript, Word and WordPerfect.

WRITE

The opposite of **read**. Putting the contents of a computer's **memory** into a **file** or **disk**.

WRITE PROJECT

See **read-only**.

WYSIWYG

Pronounced wizziwig; What-You-See-Is-What-You-Get. A (desirable) feature of **word-processors** meaning that it looks the same on the **screen** as it's going to on paper.

YES BUT../NO BUT..

The reply to any question which begins 'Is it possible...?'. Most things are possible in computing if you have enough money and/or you can wait long enough and/or you are prepared to make other compromises. For a system based entirely on the principle of on or off, yes or no, most things in computing are surprisingly vague. (cf **binary**)

ZAP

A term (American, definitely) used particularly in **spreadsheets**, meaning to delete the current contents of the spreadsheet. Similar to *delete*.